ADCHEM 2009

IFAC Symposium on

Advanced Control of Chemical Processes

July 12-15, 2009

Koç University, Istanbul, Turkey

Symposium Preprints

Part 2

Editors: Sebastian Engell

Yaman Arkun

International Program Committee

Sebastian Engell (Chair)

Technische Universität Dortmund, Germany

Jorge Mandler (Industrial Co-Chair)

Air Products, USA

IPC Coordination

Christian Sonntag, Thomas Tometzki

Technische Universität Dortmund, Germany

IPC Members

O. Abel (DE), F. Allgöwer (DE), J. Alvarez (MX), H. Arellano-Garcia (DE), T. Backx (NL), T. Badgwell (US), R. Berber (TR), L.Biegler (US), D. Bonvin (CH), R. Braatz (US), M. Çamurdan (TR), T. Chai (CN), M. Chiu (SG), P. Christofides (US), A. Çınar (US), I. Craig (ZA), P. Daoutides (US), C. de Prada (ES), D. Dochain (BE), F. J. Doyle, III (US), G. Dünnebier (DE), F. Forbes (CA), B. Foss (NO), F. Gao (HK), C. Georgakis (US), M. Guay (CA), S. Hasebe (JP), K. Hoo (US), M. Hovd (NO), B. Huang (CA), C. D. Immanuel (UK), S. B. Jorgensen (DK), M. V. Kothare (US), C. Kravaris (GR), J. Lee (US), D. Lewin (IL), P. Li (DE), J. Mandler (US), J. Marchetti (AR), W. Marquardt (DE), M. Nikolaou (US), D. Odloak (BR), C. Özgen (TR), A. Palazoğlu (US), S. Park (KR), M. Perrier (CA), H. Preisig (NO), Raisch (DE), J. Rawlings (US), J. Romagnoli (US), C. Scali (IT), A. R. Secchi (BR), S. L. Shah (CA), S. Skogestad (NO), M. Soroush (US), B. Srinivasan (CA), M. Tade (AU), N. Thornhill (UK), J. O. Trierweiler (BR), M. Türkay (TR), A. Vande Wouwer (BE), E. Ydstie (US), E. S. Yoon (KR), C.-C. Yu (TW), D. Zhou (CN)

National Organizing Committee

Yaman Arkun (Chair)

Koç University, Istanbul, Turkey

NOC Members

Erdoğan Alper, Tahsin Bahar, Ridvan Berber, Mehmet Çamurdan, Devrim Kaymak, Hitay Özbay, Canan Özgen, Metin Türkay, Mustafa Türker

NOC Coordination

Nur Diğdem Burak

Koç University, Istanbul, Turkey

COPYRIGHT CONDITIONS

Contents

Performance Assessment in Closed-loop Systems

Oral Session

Multi-step Prediction Error Approach for MPC Performance Monitoring ⋆

Yu Zhao * **Jian Chu** * **Hongye Su** * **Biao Huang** **

* *State Key Lab. of Industrial Control Technology, Institute of Cyber Systems and Control, Zhejiang University, Hangzhou, China (e-mail: yzhao@iipc.zju.edu.cn).*
** *Department of Chemical and Materials Engineering, University of Alberta, Edmonton, AB, Canada, T6G 2G6 (e-mail: biao.huang@ualberta.ca)*

Abstract: Performance monitoring of model predictive control systems (MPC) has received a great interest from both academia and industry. In recent years some novel approaches for multivariate control performance monitoring have been developed without the requirement of process models or interactor matrices. Among them the prediction error approach has been shown to be a promising one, but it is k-step prediction based and may not be fully comparable with the MPC objective that is multi-step prediction based. This paper develops a multi-step prediction error approach for performance monitoring of model predictive control systems, and demonstrates its application in an industrial MPC performance monitoring and diagnosis problem.

Keywords: Multivariable control systems, Model predictive control, Performance evaluation, Performance monitoring, Prediction error methods.

1. INTRODUCTION

Since early work of Harris (1989), research on control performance assessment (CPA) has achieved a great progress and continues to be an active area. There is a great demand from industry for this research to produce practical solutions, particularly for MPC monitoring. Many algorithms in CPA including commercial software have been developed. There are several interesting reviews addressing related research achievements in different stages (Harris et al., 1999; Huang et al., 1999; Jelali , 2006; Qin , 2007).

Even with great achievements, multivariable CPA still has a number of stumbling blocks in practical applications. Recently some progress has been made towards this direction (Jelali , 2006; Huang et al., 2006). In particular, performance assessment of model predictive control (MPC) has been an interest since MPC is the most effective and widely used advanced multivariate control strategies in modern industries. With the existence of the constraints and economic optimization, the existing CPA is not directly applicable to its performance assessment (Xu et al., 2007).

For multivariable CPA to be practical, it must reduce *a priori* knowledge requirement. Traditional approaches for the multivariable CPA with minimum variance control as the benchmark need to estimate the interactor matrices, which is equivalent to knowing the process model (Huang et al., 1999) or at least the first few Markov parameter matrices. Recently, some new methods have been developed to address the multivariable CPA problems with only the input/output data (Jelali , 2006; Huang et al., 2006).

What simple index may be considered as a measure or one of the most important MPC performance measures? Consider that, if a closed-loop output is highly predictable, one should be able to do better, i.e. to compensate the predictable content by a well designed controller. This is the principle of predictive control. Should a better controller be implemented, the closed-loop output would have been less predictable. Therefore, high predictability of a closed-loop output implies high potential to improve its performance by controller re-tuning and/or re-design, or in other words, the existing controller may not have been satisfactory in terms of exploring its potential.

However, the CPA approach based on the prediction error has an equivalence to minimum variance based performance measure (Huang et al., 2008). Thus it may not be fully comparable with the MPC objective. Motivated by the prediction-error approach of (Huang et al., 2006; Zhao et al., 2008) and multi-step identification of Shook et al. (1992), this paper further develops closed-loop prediction-error measures based on multi-step prediction that is more relevant to model predictive control. Furthermore, applications of the proposed performance measures for an industrial model predictive control system are reported in this paper.

The remainder of this paper is organized as follows: Section 2 revisits the concept of prediction-error and closed-loop potentials for CPA. Section 3 introduces the

⋆ This work is supported in part by the the National Creative Research Groups Science Foundation of China (NCRGSFC: 60421002) and National Basic Research Program of China (973 Program 2007CB714000) and by the 111 Program (B07031) for visiting professorship.

multi-step prediction error. Based on it, new potential measures are defined for the MPC controller performance assessment in Section 4. This is followed by an industrial case study in Section 5 to illustrate the utility of the new performance measures. Finally the conclusion is drawn in Section 6.

2. REVISIT OF CLOSED-LOOP POTENTIAL FOR MULTIVARIATE CPA

In this section, we shall revisit the concepts of prediction error and closed-loop potentials as defined in Huang et al. (2006).

For a multivariable process, the closed-loop output driven by white noise can be described by a time series model:

$$Y_t = G_{cl} a_t \tag{1}$$

where G_{cl} is the time series model and a_t is white noise with mean zero and covariance Σ_a.

Transfer the above time series model to a moving average (MA) form:

$$Y_t = \sum_{k=0}^{\infty} F_k a(t-k) = F_0 a_t + F_1 a_{t-1} + \cdots + F_{i-1} a_{t-(i-1)} + F_i a_{(t-i)} + \cdots \tag{2}$$

Note that this time series model can be estimated without any a priori knowledge about the process.

With the MA model, one can obtain the optimal ith step prediction:

$$Y_{t|t-i} = F_i a_{(t-i)} + F_{i+1} a_{(t-i-1)} + \cdots \tag{3}$$

and the prediction error:

$$e_{t|t-i} = Y_t - Y_{t|t-i} = F_0 a_t + F_1 a_{t-1} + \cdots + F_{i-1} a_{t-(i-1)} \tag{4}$$

where $F_0 = I$. The covariance of the prediction error can be calculated as

$$cov(e_{t|t-i}) = F_0 \Sigma_a F_0^T + F_1 \Sigma_a F_1^T + \cdots + F_{i-1} \Sigma_a F_{i-1}^T \tag{5}$$

Define its scalar measure:

$$s_i = tr(cov(e_{t|t-i})) = tr(F_0 \Sigma_a F_0^T + \cdots + F_{i-1} \Sigma_a F_{i-1}^T) \tag{6}$$

s_i is monotonically increasing with i, as $i \to \infty$, $e_{t|t-i} \to Y_t$, and $s_\infty = tr(cov(Y_t))$. If we plot s_i versus i, the plot reflects how the prediction error increases with the prediction horizon.

A closed-loop potential is defined in Huang et al. (2006) as:

$$p_i = \frac{s_\infty - s_i}{s_\infty} \tag{7}$$

The closed-loop potential can be interpreted as following (Huang et al., 2006): If a deadbeat control action can be applied from time i, then the sum of squared error (SSE) can be reduced by $100 \times p_i$ percent. From stochastic view point, if i is greater than the interactor order d, it is possible that the variance of the multivariate output can be reduced by $100 \times p_i$ percent of the current variance. Since the order of the actual interactor matrix may not be known, one can check the trajectory of the closed-loop potential versus a range of possible time lag d. As s_i is monotonically increasing with i, p_i is monotonically decreasing. When $i \to 0$, $s_0 = tr(cov(Y_t - Y_{t|t})) = 0$, $p_0 = 1$. Therefore, the index p_i starts from 1 at $i = 0$ and monotonically decreases to 0 at $i \to \infty$. Larger the closed-loop potential is, more potential the control performance can be improved.

From the potential plot we can draw the conclusion whether or how much the present closed-loop has potential to improve. Furthermore, with the plot, we can compare performance of a controller between different tuning parameters.

3. CLOSED-LOOP POTENTIAL MEASURES BASED ON MULTI-STEP PREDICTION

3.1 Multi-step optimal prediction and its scalar measure

It is well-known that minimum variance control is an aggressive control and not all controllers are designed towards minimum variance performance. Therefore, in addition to the measure of the optimal i-step prediction error s_i, which is associated with minimum variance performance, we consider a control that achieves optimal prediction performance over multi-steps, i.e. over a window from N_1 to N_2, where N_1 typically equals time delay d. In this way, we consider an optimum that is not based on a single prediction point but based on multiple prediction points.

For the multi-step optimal prediction problem, the minimization of the following multi-step prediction error is of interest (Shook et al., 1992; Huang et al., 2003):

$$s_{N_1,N_2} = \frac{1}{N_p} \sum_{j=N_1}^{N_2} E[Y_{t+j} - Y_{t+j|t}]^T [Y_{t+j} - Y_{t+j|t}] \tag{8}$$

where $Y_{t+j|t}$ is an optimal j-step ahead prediction, N_1 and N_2 are the minimum and maximum prediction step, $N_p = N_2 - N_1 + 1$, and s_{N_1,N_2} is defined as the scalar measure of the optimal multi-step prediction error (from N_1 to N_2). MPC attempts to minimize the error of multi-step predictions, i.e. from the first N_1 step to the N_2 step prediction. Thus the objective function (8) is MPC relevant.

It has been shown in Huang et al. (2003) that the objective function of multi-step prediction error is equivalent to the variance of filtered one-step prediction error:

$$\begin{aligned} s_{N_1,N_2} &= \frac{1}{N_2 - N_1 + 1} \sum_{n=N_1}^{N_2} E||[Y_{t+n} - Y_{t+n|t}]||^2 \\ &= E||[F_{N_1,N_2}(z^{-1})(Y_t - Y_{t|t-1})]||^2 \end{aligned} \tag{9}$$

where the filter $F_{N_1,N_2}(z^{-1})$ is the spectral factor of the following spectrum (Huang et al., 2003):

$$L_{N_1,N_2} = \frac{1}{N_2 - N_1 + 1} \sum_{n=N_1}^{N_2} ||F_n(e^{-nj\omega})||^2 \tag{10}$$

where

$$F_n(z^{-1}) = \sum_{i=0}^{n-1} F_n z^{-i}$$

If $N_1 = 1$ and $N_2 = k$, it is easy to show that $F_{1,k}(z^{-1})$ has the following form:

$$F_{1,k}(z^{-1}) = \tilde{F}_0 + \tilde{F}_1 z^{-1} + \cdots + \tilde{F}_{k-1} z^{-k+1} \tag{11}$$

where $\tilde{F}_i$ is to be determined next.

According to Eqn. 4, the optimal one step prediction error $Y_t - Y_{t|t-1} = a_t$, i.e. white noise. Thus

$$\begin{aligned} s_{1,k} = & \quad E[F_{1,k}(z^{-1})a_t]^T[F_{1,k}(z^{-1})a_t] \\ = & tr\{[(\tilde{F}_0 + \tilde{F}_1 z^{-1} + \cdots + \tilde{F}_{k-1}z^{-k+1})a_t]^T \\ & [(\tilde{F}_0 + \tilde{F}_1 z^{-1} + \cdots + \tilde{F}_{k-1}z^{-k+1})a_t]\} \end{aligned}$$

which can be further written as

$$s_{1,k} = tr(\tilde{F}_0\Sigma_a\tilde{F}_0^T + \tilde{F}_1\Sigma_a\tilde{F}_1^T + \cdots + \tilde{F}_{k-1}\Sigma_a\tilde{F}_{k-1}^T) \quad (12)$$

In the next two sections we will derive univariate and multivariate expressions of the optimal multi-step prediction error, respectively.

3.2 The Univariate Process

For the univariate process, the terms F_i and $\tilde{F}_i$ are both scalars (hence we use f_i and $\tilde{f}_i$ to stand for the scalar values), so the scalar prediction error measures can be simplified to the following forms:

$$s_k = (f_0^2 + f_1^2 + \cdots + f_{k-1}^2)\sigma_a^2 \quad (13)$$

$$s_{1,k} = (\tilde{f}_0^2 + \tilde{f}_1^2 + \cdots + \tilde{f}_{m-1}^2)\sigma_a^2 \quad (14)$$

When $k = 1$, by definition, $s_{1,1}$ is the variance of one step prediction error; thus

$$s_1 = s_{1,1}$$

When $k = 2$, the following result could be obtained:

$$s_{1,2} = (\tilde{f}_0^2 + \tilde{f}_1^2)\sigma_a^2 = \frac{1}{2}(2f_0^2 + f_1^2)\sigma_a^2 = \frac{1}{2}(s_1 + s_2)$$

Similarly, when $N_1 = 1, N_2 = k$, we have

$$\begin{aligned} s_{1,k} &= \frac{\sigma_a^2}{k}\{kf_0^2 + (k-1)f_1^2 + \cdots + f_{k-1}^2\} \\ &= \frac{1}{k}\sum_{i=1}^{k} s_i \end{aligned} \quad (15)$$

Thus

$$\begin{aligned} s_{k,m} = & \frac{1}{m-k+1}[(m-k+1)(f_0^2 + f_1^2 + \cdots + f_{k-1}^2) \\ & + (m-k)f_k^2 + \cdots + f_{m-1}^2]\sigma_a^2 \end{aligned}$$

Proposition 1. For a univariate control loop, the measure of optimal multi-step prediction error from k to m ($s_{k,m}$) is no smaller than that of the optimal k-step prediction error (s_k), and the two measures are asymptotically equal, namely

$$s_{k,m} - s_k \geq 0 \quad (16)$$

$$\lim_{k\to\infty}\{s_{k,m} - s_k\} = 0 \quad (17)$$

Proof.

Recall that the measures of the optimal k-step prediction error and optimal multi-step prediction error are respectively:

$$s_k = (f_0^2 + f_1^2 + \cdots + f_{k-1}^2)\sigma_a^2 \quad (18)$$

and

$$\begin{aligned} s_{k,m} = & \frac{1}{m-k+1}[(m-k+1)(f_0^2 + f_1^2 + \cdots + f_{k-1}^2) \\ & + (m-k)f_k^2 + \cdots + f_{m-1}^2]\sigma_a^2 \end{aligned} \quad (19)$$

Thus

$$\begin{aligned} s_{k,m} - s_k = & \frac{1}{m-k+1}[(m-k+1)(f_0^2 + f_1^2 + \cdots + f_{k-1}^2) \\ & + (m-k)f_k^2 + \cdots + f_{m-1}^2]\sigma_a^2 - (f_0^2 + f_1^2 + \cdots + f_{k-1}^2)\sigma_a^2 \\ \triangleq & \frac{1}{N_p}[(N_p-1)f_k^2 + (N_p-2)f_{k+1}^2 + \cdots + f_{m-1}^2]\sigma_a^2 \quad (20) \\ \geq & 0 \end{aligned}$$

where $N_p = m - k + 1$.

Consider a stable closed-loop response:

$$\begin{aligned} y_t = G_{cl}(z^{-1};\theta)e_t &= \frac{B(z^{-1})}{A(z^{-1})}e_t \\ &= \frac{b_0 + b_1 z^{-1} + \cdots + b_m z^{-m}}{a_0 + a_1 z^{-1} + \cdots + a_n z^{-n}}e_t \end{aligned} \quad (21)$$

Write the above transfer function in the zero-pole form

$$y_t = \frac{b_0(1-\beta_1 z^{-1})(1-\beta_2 z^{-1})\cdots(1-\beta_m z^{-1})}{(1-\alpha_1 z^{-1})(1-\alpha_2 z^{-1})\cdots(1-\alpha_n z^{-1})}e_t \quad (22)$$

where $|\alpha_i| < 1$.

Partial fraction expansion of Eqn. (22) yields

$$\begin{aligned} y_t &= (\frac{c_1}{1-\alpha_1 z^{-1}} + \frac{c_2}{1-\alpha_2 z^{-1}} + \cdots + \frac{c_n}{1-\alpha_n z^{-1}})e_t \\ &= \sum_{p=1}^{n} c_p(1 + \alpha_p z^{-1} + \alpha_p^2 z^{-1} + \cdots)e_t \\ &\triangleq (f_0 + f_1 z^{-1} + \cdots + f_i z^{-i}\cdots)e_t \end{aligned} \quad (23)$$

where

$$f_i = \sum_{p=1}^{n} c_p\alpha_p^i \quad (24)$$

So, the *ith* term of Eqn. (18) can be calculated as

$$\begin{aligned} f_i^2 &= (\sum_{p=1}^{n} c_p\alpha_p^i)^2 \\ &= \sum_{p=1}^{n} c_p^2\alpha_p^{2i} + 2\sum_{p=1}^{n-1}\sum_{q=p+1}^{n} c_p c_q(\alpha_p\alpha_q)^i \end{aligned} \quad (25)$$

According to Eqn. (20)

$$s_{k,m} - s_k = \frac{1}{N_p}\sum_{i=k}^{m-1}(m-i)f_i^2\sigma_a^2 \quad (26)$$

Substituting Eqn. (25) in Eqn. (26), we obtain that

$$\begin{aligned} & s_{k,m} - s_k \\ = & \frac{\sigma_a^2}{N_p}\sum_{i=k}^{m-1}(m-i)(\sum_{p=1}^{n} c_p^2\alpha_p^{2i} + 2\sum_{p=1}^{n-1}\sum_{q=p+1}^{n} c_p c_q(\alpha_p\alpha_q)^i) \\ = & \frac{\sigma_a^2}{N_p}(m\sum_{p=1}^{n} c_p^2\sum_{i=k}^{m-1}\alpha_p^{2i} - \sum_{p=1}^{n} c_p^2\sum_{i=k}^{m-1} i\alpha_p^{2i} \\ & + \sum_{p=1}^{n-1}\sum_{q=p+1}^{n} 2mc_p c_q\sum_{i=k}^{m-1}(\alpha_p\alpha_q)^i \\ & - \sum_{p=1}^{n-1}\sum_{q=p+1}^{n} 2c_p c_q\sum_{i=k}^{m-1} i(\alpha_p\alpha_q)^i) \end{aligned} \quad (27)$$

where the terms $\sum_{i=k}^{m-1}\alpha_p^{2i}$, $\sum_{i=k}^{m-1} i\alpha_p^{2i}$, $\sum_{i=k}^{m-1}(\alpha_p\alpha_q)^i$ and $\sum_{i=k}^{m-1} i(\alpha_p\alpha_q)^i$ can be determined respectively as

$$\sum_{i=k}^{m-1}\alpha_p^{2i}=\frac{\alpha_p^{2k}(1-\alpha_p^{2(m-k)})}{1-\alpha_p^2} \tag{28}$$

$$\sum_{i=k}^{m-1}i\alpha_p^{2i}=\frac{\alpha_p^{2k}(1-\alpha_p^{2(m-k)})}{(1-\alpha_p^2)^2}+\frac{(k-1)\alpha_p^{2k}-(m-1)\alpha_p^{2m}}{(1-\alpha_p^2)} \tag{29}$$

$$\sum_{i=k}^{m-1}(\alpha_p\alpha_q)^i=\frac{(\alpha_p\alpha_q)^k(1-(\alpha_p\alpha_q)^{m-k})}{1-\alpha_p\alpha_q} \tag{30}$$

$$\sum_{i=k}^{m-1}i(\alpha_p\alpha_q)^i=\frac{(\alpha_p\alpha_q)^k(1-(\alpha_p\alpha_q)^{m-k})}{(1-\alpha_p\alpha_q)^2}+\frac{(k-1)(\alpha_p\alpha_q)^k-(m-1)(\alpha_p\alpha_q)^m}{(1-\alpha_p\alpha_q)} \tag{31}$$

Substituting the above four equations in Eqn. (27) yields

$$\begin{aligned}&s_{k,m}-s_k\\&=\frac{\sigma_a^2}{N_p}(\sum_{p=1}^{n}c_p^2(-\frac{\alpha_p^{2k}(1-\alpha_p^{2(m-k)})}{(1-\alpha_p^2)^2}+\frac{N_k\alpha_p^{2k}-\alpha_m^2}{1-\alpha_p^2})\\&\quad+\sum_{p=1}^{n-1}\sum_{q=p+1}^{n}2c_pc_q(-\frac{(\alpha_p\alpha_q)^k(1-(\alpha_p\alpha_q)^{m-k})}{(1-\alpha_p\alpha_q)^2}\\&\quad+\frac{N_k(\alpha_p\alpha_q)^k-(\alpha_p\alpha_q)^m}{1-\alpha_p\alpha_q}))\\&\triangleq\sum_{p=1}^{n}c_p^2\times Sum1+\sum_{p=1}^{n-1}\sum_{q=p+1}^{n}(2c_pc_q)\times Sum2\end{aligned} \tag{32}$$

where

$$Sum1=\frac{\sigma_a^2}{N_p}[-\frac{\alpha_p^{2k}(1-\alpha_p^{2(m-k)})}{(1-\alpha_p^2)^2}+\frac{N_p\alpha_p^{2k}-\alpha_m^2}{1-\alpha_p^2}]$$

$$Sum2=\frac{\sigma_a^2}{N_p}[-\frac{(\alpha_p\alpha_q)^k(1-(\alpha_p\alpha_q)^{m-k})}{(1-\alpha_p\alpha_q)^2}+\frac{N_p(\alpha_p\alpha_q)^k-(\alpha_p\alpha_q)^m}{1-\alpha_p\alpha_q}]$$

When $k\to\infty$, $m\to\infty$ since $m\geq k$. Let $m-k=P\geq 0$. Consequently, $N_p=m-k+1=P+1$. The limits of $Sum1$ and $Sum2$ can be obtained:

$$\begin{aligned}\lim_{k\to\infty}Sum1&=\lim_{k\to\infty}[-\frac{\alpha_p^{2k}(1-\alpha_p^{2(m-k)})}{N_p(1-\alpha_p^2)^2}+\frac{N_p\alpha_p^{2k}-\alpha_p^{2m}}{N_p(1-\alpha_p^2)}]\sigma_a^2\\&=0\end{aligned} \tag{33}$$

As $|\alpha_i|<1$, obviously $|\alpha_p\alpha_q|<1$. Similarly,

$$\begin{aligned}&\lim_{k\to\infty}Sum2\\&=\lim_{k\to\infty}(-\frac{(\alpha_p\alpha_q)^k(1-(\alpha_p\alpha_q)^{m-k})}{N_p(1-\alpha_p\alpha_q)^2}+\frac{N_p(\alpha_p\alpha_q)^k-(\alpha_p\alpha_q)^m}{N_p(1-\alpha_p\alpha_q)})\sigma_a^2\\&=0\end{aligned} \tag{34}$$

Consequently,

$$\begin{aligned}&\lim_{k\to\infty}\{s_{k,m}-s_k\}\\&=\sum_{p=1}^{n}c_p^2\times Sum1+\sum_{p=1}^{n-1}\sum_{q=p+1}^{n}(2c_pc_q)\times Sum2\\&=0\end{aligned} \tag{35}$$

3.3 The multivariate process

Following a similar procedure as that for the univariate process, the following measure of optimal multi-step prediction error can be derived:

$$\begin{aligned}s_{k,\,m}&=tr(\tilde{F}_0\Sigma_a\tilde{F}_0^T+\tilde{F}_1\Sigma_a\tilde{F}_1^T+\cdots+\tilde{F}_{m-1}\Sigma_a\tilde{F}_{m-1}^T)\\&=\sum_{p=0}^{k-1}\sum_{j=1}^{N}\sum_{i=1}^{N}f_{ijp}^2\sigma_j^2+\sum_{p=k}^{m-1}\frac{m-p}{m-k+1}\sum_{j=1}^{N}\sum_{i=1}^{N}f_{ijp}^2\sigma_j^2\end{aligned} \tag{36}$$

A same proposition as in the univariate case can be proved, but is omitted in this shorter version due to space limit.

4. CLOSED-LOOP POTENTIALS BASED ON THE MULTI-STEP PREDICTION

Based on the multi-step prediction error derived in the above section, the following closed-loop potential measure is defined for performance assessment of MPC:

$$p_{k,\,m}=\frac{s_{\infty,\,\infty+N_p}-s_{k,\,m}}{s_{\infty,\,\infty+N_p}} \tag{37}$$

where $N_p=P+1$ and P represents the prediction horizon. It can be shown that $s_{\infty,\,\infty+N_p}=s_\infty=trace\{cov(Y_t)\}$.

Here we use the multi-step prediction error scalar measure $s_{k,\,m}$ instead of the k-step prediction error scalar measure s_k of Huang et al. (2006) to derive closed-loop potential. As has been proven in the last section, with a fixed prediction horizon P, i.e. $m-k=P=N_p-1$, the scalar measure $s_{k,\,m}$ is monotonically increasing with k. So $p_{k,\,m}$ is monotonically decreasing. When $k=0$, $s_{k,\,m}\geq=s_0=0$, so $p_{0,\,N_p}\leq 1$. Besides, as $s_{\infty,\,\infty+N_p}\geq s_{k,\,m}$, consequently, $0\leq p_{k,\,m}\leq 1$. According to its definition we can see that the $p_{k,\,m}$ is dimensionless and in addition, the potential measure is more relevant to the MPC control strategy as it is multi-step prediction based. The actual process time delay (that corresponds to k) may not be known in practice. So the trajectory of the potential measure with a range of k will be more useful to assess the performance of the controller.

It is also desirable to know details about the performance of each output. So, the individual scalar potential measure is required. By replacing the operator $tr(\cdot)$ in Eqn. (36) with $diag(\cdot)$, the individual scalar measure is defined as:

$$s_{k,\,m}^{ind}=diag(\tilde{F}_0\Sigma_a\tilde{F}_0^T+\tilde{F}_1\Sigma_a\tilde{F}_1^T+\cdots+\tilde{F}_{m-1}\Sigma_a\tilde{F}_{m-1}^T)$$

where $s_{k,\,m}^{ind}\in\mathbb{R}^{N\times 1}$ and N represents the number of controlled variables.

The ith component of $s_{k,\,m}^{ind}$ can be obtained:

$$s_{k,\,m}^{ind}(i)=\sum_{p=0}^{k-1}\sum_{j=1}^{N}f_{jip}^2\sigma_i^2+\sum_{p=k}^{m-1}\frac{m-p}{m-k+1}\sum_{j=1}^{N}f_{jip}^2\sigma_i^2 \tag{38}$$

As a result, the individual potential measure can be defined as:

$$p_{k,\,m}^{ind}(i)=\frac{s_{\infty,\,\infty+N_p}^{ind}(i)-s_{k,\,m}^{ind}(i)}{s_{\infty,\,\infty+N_p}^{ind}(i)} \tag{39}$$

where $s_{\infty,\,\infty+N_p}^{ind}=diag(s_{\infty,\,\infty+N_p})=diag(cov(Y_t))$ and $1\leq i\leq N$.

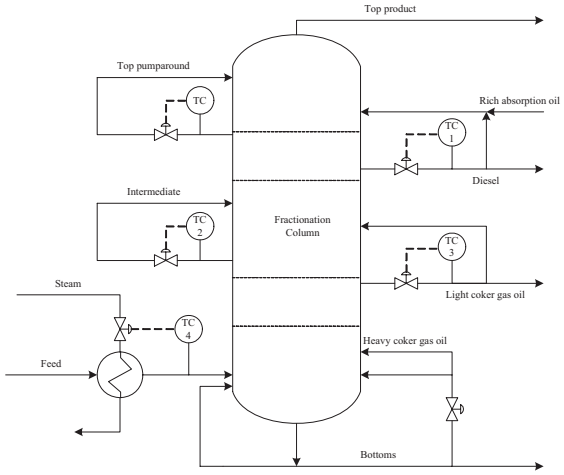


Fig. 1. Schematic diagram of the fractionation column in the delayed coking unit.

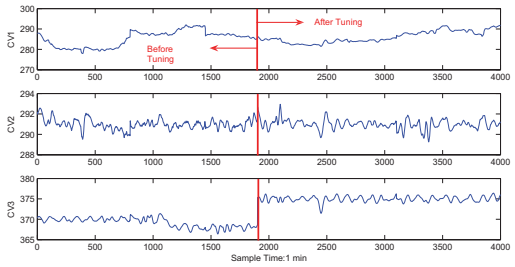


Fig. 2. Output data set under MPC controller.

5. INDUSTRIAL APPLICATION

In this section the proposed multi-step closed-loop potential measures will be applied to evaluate the performance of an industrial control system.

5.1 Process description

This is a control performance assessment and diagnosis problem for a MPC control system in a delayed coking refinery unit. Fig. 1 is a simplified process flow chart.

The control system consists of three manipulated variables (MVs), three controlled variables (CVs) and one disturbance variable (DV), the temperature of the feedstocks. A description of process variables and their corresponding tag names and the parameters for the MPC design is shown in Table 1.

Two different closed-loop operation data sets are collected with 1 min sampling interval under different MPC controller tunings as shown in Fig. 2. All the data is selected without the drum events to avoid unusual upset. The first part of the data from 1 to 1900 is selected before the controller tuning and the rest part is selected after the controller tuning.

5.2 Performance assessment

By using the proposed approach, the scalar potential measure trajectories for each data set are generated and

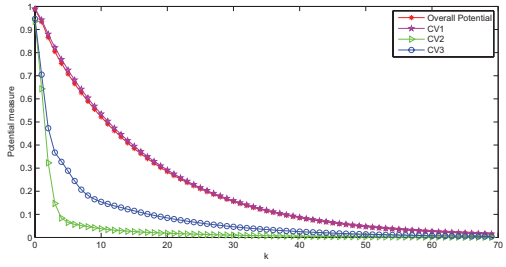


Fig. 3. Scalar potential measures of the system before controller tuning.

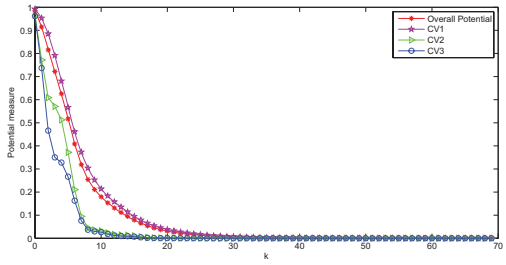


Fig. 4. Scalar potential measures of the system after controller tuning.

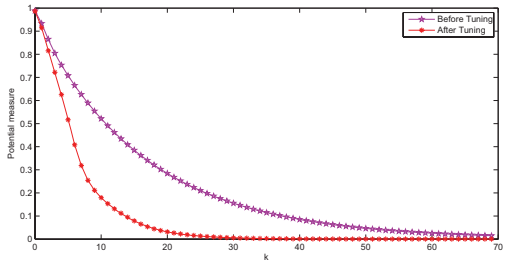


Fig. 5. Overall scalar potential measures of the system under different controller tunings.

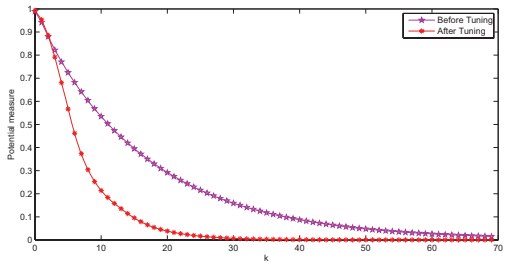


Fig. 6. Individual scalar potential measures of the system under different controller tunings of CV1.

shown in Fig. 3 and 4, respectively. The comparisons of the overall potential and individual potential for each CV are displayed from Fig. 5 to Fig. 8.

With these figures, the following performance analysis conclusions can be obtained:

Table 1. List of process variables and their corresponding tag names and parameters for MPC design.

No.	Tag	Weight	Horizon	Operation Range
CV1	Temperature of diesel	10	20	275-295
CV2	Temperature in the intermediate	1	20	285-295
CV3	Temperature of light coker gas oil	1	20	360-380
MV1	Valve opening of diesel	0.1	3	0-100
MV2	Valve opening of intermediate reflux	0.1	3	0-100
MV3	Valve opening of light coker gas oil reflux	0.1	3	0-100

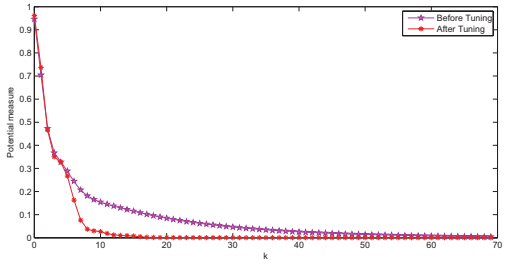


Fig. 7. Individual scalar potential measures of the system under different controller tunings of CV2.

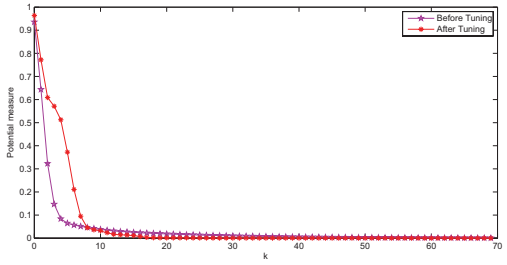


Fig. 8. Individual scalar potential measures of the system under different controller tunings of CV3.

- According to Fig 3, before the controller tuning, the overall scalar potential measure trajectory converges slowly and CV1 contributes the more potential to the overall potential than the other two CVs.
- According to Fig 4, after the controller tuning, the overall scalar potential measure trajectory converges fast and the three individual trajectories come close to each other, although the trajectory of CV1 still lies above the other two.
- According to Fig 5, there is significant improvement of the system's performance after the tuning as there is less potential after the controller tuning than that before tuning.
- According to Fig 6, 7 and 8, performance of both CV1 and CV2 is improved after the controller tuning; however, performance of CV3 is degraded after the tuning.
- The above results indicate that the improvement of the overall performance comes from the improvement of CV1 and CV2 but at some cost of CV3. As CV1 is the most important quality variable (the weight of CV1 is larger than those of the other two in Table 1.), it is worth to improve its performance by slightly deteriorating the performance of CV3.

In summary, after the controller tuning, there is significant improvement of the system's performance.

6. CONCLUSION

The closed-loop potentials are promising measures of model predictive control performance. However, they have certain limitations as they are originally defined. In this paper, new closed-loop potentials are proposed. The proposed performance potentials are multi-step prediction based and thus MPC relevant. Regardless of the dimension of the plant, the closed-loop potentials can be easily calculated, which facilitates the implementation, visualization, and interpretation. Industrial application demonstrates powerfulness of the the proposed performance measures.

REFERENCES

Harris, T.J., C.T. Seppala, and L.D. Desborough. A review of performance monitoring and assessment techniques for univariate and multivariate control systems. *Journal of Process Control*, 9:1-17, 1999.

Harris, T.J. Assessment of closed loop performance. *Can.J.Chem.Eng.*, 67:856-861, 1989.

Huang, B. and R. Kadali. *Dynamic Modeling, Predictive Control and Performance Monitoring: A Data-driven Subspace Approach*. Verlag: Springler, 2008.

Huang, B., S.X. Ding, and N. Thornhill. Alternative solutions to multi-variate control performance assessment problems. *Journal of Process Control*, 16:457-471, 2006.

Huang, B., A. Malhotra, and E.C. Tamayo. Model predictive control relevant identification and validation. *Chemical Engineering Science*, 58:2389-2401, 2003.

Huang, B. and S.L. Shah. *Performance Assessment of Control Loops: Theory and applications*. Springer: New York, 1999.

Jelali, M. An overview of control performance assessment technology and industrial applications. *Control Engineering Practice*, 14:441-466, 2006

Qin, S.J. Recent developments in multivariable controller performance monitoring. *Journal of Process Control*, 17:221-227, 2007.

Shook, D.S., C. Mohtadi and S.L. Shah. A Control-Relevant Identification Strategy for GPC. *IEEE Trans. AC*, 37, 1992.

Xu, F., B Huang, and S. Akande. Performance Assessment of Model Predictive Control for Variability and Constraint Tuning. *Ind. Eng. Chem. Res.*, 46:1208-1219, 2007.

Zhao, Y., Y. Gu, H. Su, B. Huang. Extended prediction error approach for MPC performance monitoring and industrial applications. *17th IFAC World Congress*, Seoul, 2008.

Valve friction and nonlinear process model closed-loop identification

Rodrigo A. Romano * **Claudio Garcia** **

* *Polytechnic School, University of São Paulo, São Paulo, Brazil (Tel: +55 11-30911893, e-mail: rodrigo.romano@poli.usp.br)*
** *Polytechnic School, University of São Paulo, São Paulo, Brazil (Tel: +55 11-30915648, e-mail: clgarcia@lac.usp.br)*

Abstract: Friction in control valves and inadequate controller tuning are two of the major sources of control loop performance degradation. As friction models are needed to diagnose abnormal valve operation or to compensate such undesirable effects, process models play an essential role in controller design. This paper extends existing optimization-based methods that jointly identify the process and friction model parameters, so that a nonlinear process model structure is considered. The procedure is based on data generated from closed-loop experiments with an external test signal. A simulation example indicates that the method accurately quantifies the valve friction, the process dynamics and the nonlinear steady state characteristics, even when the system is subjected to different level of disturbances.

Keywords: Control valves; Nonlinear models; Identification algorithms; Friction.

1. INTRODUCTION

Among several process variability sources (e.g., inadequate controller structure/ tuning, equipment malfunction, poor design, lack of maintenance) valve friction is supposed to be one of the most prevalent (Desborough and Miller, 2001). For this reason, friction quantification methods are highly desirable, since they can be applied in the development of model-based compensators or to diagnose valves that need repair. Moreover, quantification methods based only on controller output (*op*) and process output (*pv*) measurements from closed-loop experiments, are preferable for practical reasons.

Choudhury et al. (2004) dealt with friction quantification by means of the *pv-op* plot, but the results produced by this technique depend on the controller tuning. In a method proposed by Srinivasan et al. (2005), an optimization approach is used to jointly estimate the process dynamics and the friction model parameters. This method can be seen as a Hammerstein model identification, since the valve friction is treated as a nonlinear static block ($\mathcal{N}$) followed by a linear dynamic block ($\mathcal{L}$) that represents the process. As the process dynamics is also estimated, the joint procedure previously mentioned can be used for controller retuning. However, in that work, an inappropriate friction model structure that is unable to reproduce important sticky valve characteristics is employed. In a recent work (Choudhury et al., 2008), this drawback was eliminated through the adoption of another friction model structure.

An additional extension to the method originally proposed by Srinivasan et al. (2005) is to model the process with a Wiener structure (Figure 1), built up with a linear dynamic block connected to a nonlinear static function ($\mathcal{L} \rightarrow \mathcal{N}$). In this approach the Hammerstein structure is

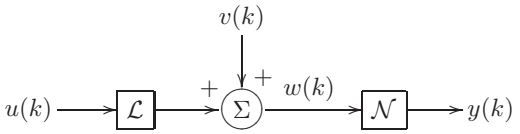


Fig. 1. Wiener model structure with nonlinear disturbance.

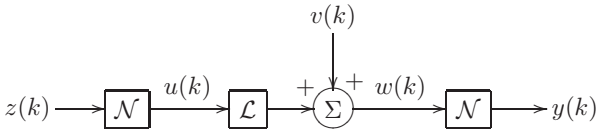


Fig. 2. Hammerstein-Wiener model structure with nonlinear disturbance.

extended to a Hammerstein-Wiener one ($\mathcal{N} \rightarrow \mathcal{L} \rightarrow \mathcal{N}$), i.e., the valve friction is associated with the first nonlinear block, while the remainder blocks represent the process. The Hammerstein-Wiener structure is shown in Figure 2.

This extension intends to provide some features: (*i*) to avoid that process nonlinearities be erroneously incorporated in the friction model, (*ii*) to prevent bias problems in the process model identification and (*iii*) to turn the estimation method suitable to wider operating ranges.

This work proposes a procedure to jointly estimate nonlinear process dynamics and friction model parameters from closed-loop experiments. Actually, it is an extension from previous works (Choudhury et al., 2008; Srinivasan et al., 2005). The paper is organized as follows: the structure that models the valve friction is described in section 2. The parameterization of the nonlinear process, as well as an estimation algorithm suitable for closed-loop data are treated in section 3. The friction and process model joint estimation procedure is presented in section 4. This

procedure is tested through a simulated example in section 5. At last, the conclusions are drawn.

2. VALVE FRICTION MODEL

Several friction models were evaluated using ISA standard tests in Garcia (2008). The best trade-off between accuracy and simplicity was achieved by the data-driven model proposed by Kano et al. (2004). This is a modified version of the model employed in the friction quantification algorithm proposed in Choudhury et al. (2008) and it is characterized by two parameters: S that represents the cummulative input signal $z(k)$ amplitude change necessary to revert the valve movement direction and J that is the size of the stem slip, also referred as slip-jump, observed when the valve starts to move.

Besides the parameters S and J, the friction model uses three auxiliar variables: *stp* that indicates if the valve is moving ($stp = 0$) or if it is stuck ($stp = 1$), z_s that is updated with $z(k)$ every time the valve sticks and $d = \pm 1$ that denotes the direction of the friction force.

The relationship between the command signal $z(k)$ and the valve stem position $u(k)$ is described in the flowchart shown in Figure 3. After testing whether the valve stopped, so that z_s and *stp* are eventually updated, a new value is assigned to $u(k)$ if: (i) the valve is moving ($stp = 0$), (ii) the valve changes its direction and overcomes S or (iii) the valve moves in the same direction and overcomes J. On the contrary, the position remains the same.

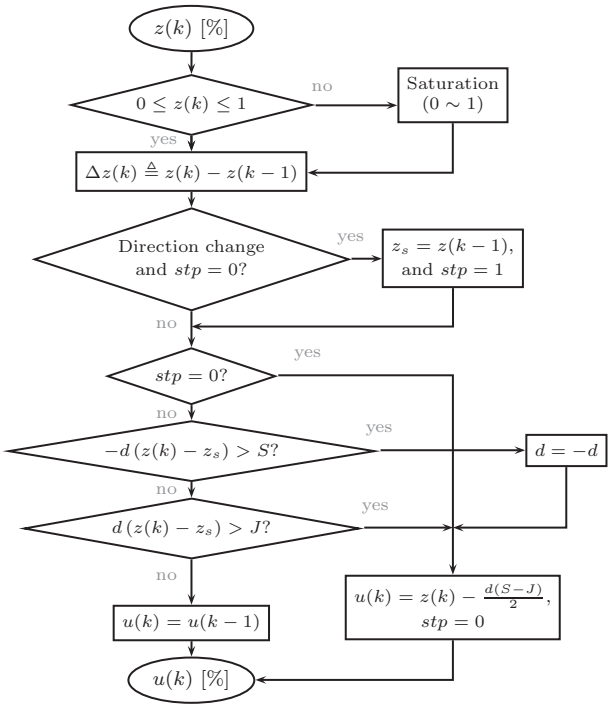


Fig. 3. Flowchart of the data-driven model parameterized by S and J (Kano et al., 2004).

3. NONLINEAR PROCESS MODEL

3.1 Model parameterization

Consider the Wiener model depicted in Figure 1, where the input signal is denoted by $u(k)$, the output signal by $y(k)$ and $v(k)$ represents the process disturbances. Notice that $v(k)$ is applied before the nonlinear block. In this case, the disturbances are also subject to the process nonlinearity. This scheme, proposed by Zhu (1999), is more realistic from a process operation point of view.

The linear dynamic block can be represented by a rational transfer function of order n:

$$G(q) = \frac{b_1 q^{-1} + \ldots + b_n q^{-n}}{1 + a_1 q^{-1} + \ldots + a_n q^{-n}} = \frac{B(q)}{A(q)} \tag{1}$$

where q^{-1} is the backward operator.

When prior knowledge about the process nonlinearity is not available, piecewise polynomials of third degree (cubic spline) provide advantages in respect of polynomials and piecewise linear functions to model the nonlinear block. For a set of m different knots:

$$w_{\min} = w_1 < w_2 < \ldots < w_{m-1} < w_m = w_{\max} \tag{2}$$

A cubic spline can be expressed by (Lancaster and Šalkauskas, 1986):

$$\begin{aligned} y(k) &= f\left(w(k)\right) \\ &= \sum_{i=2}^{m-1} \xi_i \left|w(k) - w_i\right|^3 + \xi_m + \xi_{m+1} w(k) \end{aligned} \tag{3}$$

where $\Xi \triangleq \left(\xi_2, \ldots, \xi_{m+1}\right)^T$ is the cubic spline parameter vector and $w(k)$ denotes the Wiener model intermediate signal.

3.2 Wiener model parameter estimation

In the closed-loop identification of Wiener models, the prediction error approach yields unbiased estimates, provided the process and the disturbance models are built simultaneously and the process model contains at least a delay of one sampling period (Forssell, 1999). To satisfy this condition, the disturbance term is modeled using an Auto Regressive Moving Average (ARMA) structure:

$$\begin{aligned} v(k) &= H(q)e(k) = \frac{C(q)}{D(q)} e(k) \\ &= \frac{1 + c_1 q^{-1} + \ldots + c_{n_c} q^{-nc}}{1 + d_1 q^{-1} + \ldots + d_{n_d} q^{-nd}} e(k) \end{aligned} \tag{4}$$

where $e(k)$ is white noise with zero mean and variance σ^2.

Suppose that the function which describes the process nonlinearity is monotonic and invertible. Hence, analogously to (3), the inverse of the process nonlinearity $f^{-1}(\cdot)$ can be denoted by:

$$w(k) = \sum_{i=2}^{p-1} \gamma_i \left|y(k) - y_i\right|^3 + \gamma_p + \gamma_{p+1} y(k) \tag{5}$$

Furthermore, as the intermediate signal $w(k)$ is unmeasurable, the gain of the Wiener model can be arbitrarily distributed between the dynamic and the static block. For this reason, the constraint $\gamma_{p+1} = 1$ is introduced in (5), so that the parameters can be uniquely determined.

The Wiener model parameters can be obtained from the minimization of the prediction error criterion:

$$V = \sum_k \left(H^{-1}(q)\left(w(k) - G(q)u(k)\right)\right)^2 \tag{6}$$

In order to estimate the Wiener and disturbance model parameters, besides the assumption that the process nonlinearity is invertible, the algorithm considers that the process is open loop stable. Both assumptions are commonly found in many practical situations, e.g., CSTRs, distillation columns and pH neutralization processes. Hence, $G(q)$ can be approximated by a finite impulse response (FIR) model, so that the intermediate signal is expressed by:

$$w(k) = \beta_1 u(k-1) + \ldots + \beta_r u(k-r) + v(k) \tag{7}$$

For more compact notation, consider the regression $\psi(k)$ and the parameter θ vectors:

$$\psi(k) \triangleq \Big(-|y(k) - y_2|^3, \ldots, -|y(k) - y_{p-1}|^3, -1, u(k-1), \ldots, u(k-r)\Big)^T \tag{8}$$

$$\theta \triangleq (\gamma_2, \ldots, \gamma_{p-1}, \gamma_p, \beta_1, \ldots, \beta_r)^T \tag{9}$$

Considering (8) and (9), (6) can be rewritten as:

$$V = \sum_k \left(H^{-1}(q)\left(y(k) - \psi^T(k)\theta\right)\right)^2 \tag{10}$$

Since the criterion (10) is a nonlinear least-squares problem, the following algorithm is employed to calculate $G(q)$, $f(\cdot)$ and $H(q)$:

Algorithm 1. Wiener and ARMA disturbance model parameter estimate.

i. Initialize the disturbance model $H(q)$ with:

$$\hat{C}(q) = \hat{D}(q) = 1 \tag{11}$$

ii. Calculate filtered version of the output and the regression vectors:

$$y_f(k) = \frac{D(q)}{C(q)} y(k)$$

$$\psi_f(k) = \frac{D(q)}{C(q)} \psi(k)$$

iii. Estimate the parameter vector θ from:

$$\hat{\theta} = \left(\sum_k \psi_f(k)\psi_f^T(k)\right)^{-1} \left(\sum_k \psi_f(k) y_f(k)\right) \tag{12}$$

iv. Calculate the residuals $\zeta(k)$ of the Wiener model obtained from the previous step:

$$\zeta(k) = y(k) - \psi^T(k)\hat{\theta} \tag{13}$$

v. Estimate an ARMA model for $\zeta(k)$, i.e., a filter to uncorrelate the residuals:

$$\hat{D}(q)\zeta(k) = \hat{C}(q)e(k) \tag{14}$$

vi. While convergence of $\hat{H}(q)$ does not occur, go to step (ii). Otherwise, go to the next step.

vii. The parameters of $A(q)$ and $B(q)$, defined in (1), are estimated by minimizing the error between the outputs of the FIR model and the transfer function $G(q)$:

$$V_{red} = \sum_k \left(\sum_{i=1}^{r} \hat{\beta}_i u(k-i) - \frac{B(q)}{A(q)} u(k)\right)^2 \tag{15}$$

viii. The nonlinear block parameter vector Ξ estimate is given by:

$$\hat{\Xi} = \arg\min_{\Xi} \sum_k \left(y(k) - \phi^T(k)\Xi\right)^2 \tag{16}$$

where:

$$\phi(k) \triangleq \left(|\hat{w}(k) - w_2|^3, \ldots, |\hat{w}(k) - w_{m-1}|^3, 1, \hat{w}(k)\right)^T$$

$$\hat{w}(k) \triangleq \hat{f}^{-1}\left(y(k)\right)$$

Correspondingly to the iterative calculation of θ, the linear model reduction (15) and the nonlinear function determination (16) are formulated as linear least squares problems. For this reason, the procedure is considered to be numerically simple and suitable for practical situations.

4. FRICTION AND PROCESS MODEL JOINT IDENTIFICATION ALGORITHM

Consider the process control loop depicted in Figure 4. Since in most of the practical situations only the controller output and the process output are known, the problem to be treated is to identify the friction and process model by means of $z(k)$ and $y(k)$.

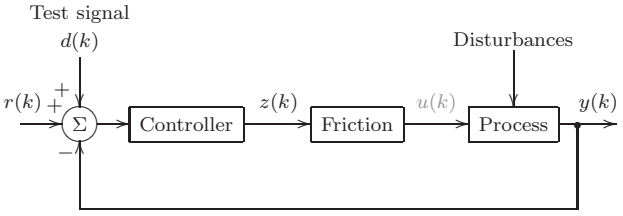


Fig. 4. Process control loop subject to valve friction.

In this work, the friction block is represented by the data-driven model of section 2, while the process dynamics is modeled by a Wiener structure. These parameterizations originate the control loop model shown in Figure 5.

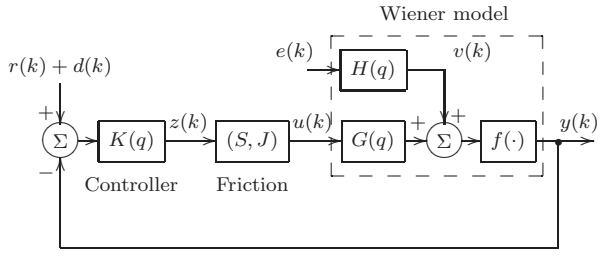


Fig. 5. Control loop where the valve friction and the process are modeled by a Hammerstein-Wiener structure.

In a first moment, suppose that the friction model parameters S and J are known. With this in mind, as the

controller output $z(k)$ is considered to be measurable, it is possible to estimate $u(k)$ with:

$$\hat{u}(k) = \mathcal{F}\left(z(k), \hat{u}(k-1), S, J\right) \tag{17}$$

where $\mathcal{F}(\cdot)$ is the nonlinear transformation described in the flowchart of Figure 3. Hence, the Wiener model parameters can be estimated, using the measured output $y(k)$ and $\hat{u}(k)$ instead of $u(k)$, by means of the algorithm presented in section 3. However, S and J are unknown. To deal with this fact, the following algorithm is proposed:

***Algorithm** 2.* Algorithm that simultaneously estimate the parameters of the friction and nonlinear process model.

i. Generate a set of candidate values for the pair (S, J). Two aspects are considered in order to restrict the set of candidate values: (a) the behavior of most real valves is reproduced by the data-driven model with $\max(J) \leq S$; (b) it is obvious that without stem movement, to estimate the valve friction is an impossible task. If an appropriate excitation $d(k)$ is employed, it is reasonable to consider that the stem velocity reversions are produced by the test signal. Therefore, the controller output imposes an upper bound for S:

$$\max(S) < \max(z(k)) - \min(z(k)) \tag{18}$$

Such constraints yield the geometric locus shown in Figure 6.

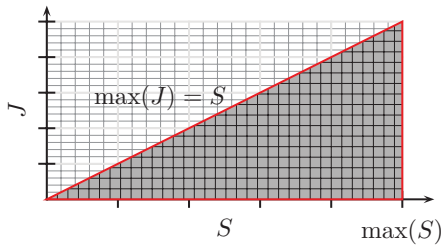


Fig. 6. Geometric locus of the friction model parameter candidate values.

ii. Choose a pair (S_i, J_j) from the set described in the previous step.

iii. Calculate the sequence of values $\hat{u}(k)$ from (17).

iv. Estimate the process model parameters using algorithm *1*, described in section 3.

v. Compute the prediction error of the intermediate signal $w(k)$ through the criterion:

$$\mathcal{C} = \sum_{k} \left(\hat{H}^{-1}(q) \left(\hat{f}^{-1}(y(k)) - \hat{G}(q)\, \hat{u}(k) \right) \right)^2 \tag{19}$$

vi. Until all the candidate values have been tested, back to step (ii). Otherwise, the values of S, J, $G(q)$ and $f(\cdot)$ are supposed to be the ones for which $\mathcal{C}$ is minimum.

Furthermore, note in Figure 4 that a test signal $d(k)$ is introduced into the set-point. Although external interferences are highly undesirable, the test signal guarantees sufficiently informative experiments. A well-known result from the closed-loop identification literature (Ljung, 1999) is that prediction error approach is not consistent if the data have been collected exclusively under feedback. In fact, the variance on the parameter estimate increases with disturbances and decreases the higher $d(k)$ is.

5. SIMULATIONS

To verify the applicability of the friction and process joint identification algorithm, the process loop in Figure 4 is simulated with a PI controller $C(q)$, a process dynamics reproduced by a continuous linear dynamic model $G(s)$ followed by a nonlinear block $f(w(k))$ and a disturbance $v(k)$ given by:

$$C(q) = \frac{0.5\left(1 - 0.5q^{-1}\right)}{\left(1 - q^{-1}\right)}$$

$$G(s) = \frac{5}{(0.5s+1)(s+1)(10s+1)}$$

$$y(k) = f\left(w(k)\right) = \frac{w(k)}{\sqrt{0.1 + 0.9w^2(k)}}$$

$$v(k) = \frac{\rho}{1 - 2.65q^{-1} + 2.335q^{-2} - 0.684q^{-3}} e(k)$$

The simulated friction model parameters are $S = 10\%$ and $J = 2\%$. The algorithm is tested in two distinct situations: low and high disturbances. In the first case, ρ is adjusted so that the disturbance level in $y(k)$ is 1.44% (in variance), while in the high disturbance scenario $v(k)$ provides a ratio of 12.5%.

A randomly switched multi-level signal, GMN (see Zhu, 2001), with average switch time of 25 sampling intervals and amplitude uniformly distributed between $[-0.15, 0.15]$ is applied in $d(k)$. The set-point $r(k)$ is fixed in 0.75. The input-output data of the high disturbance situation, as well as the excitation signal are shown in Figure 7.

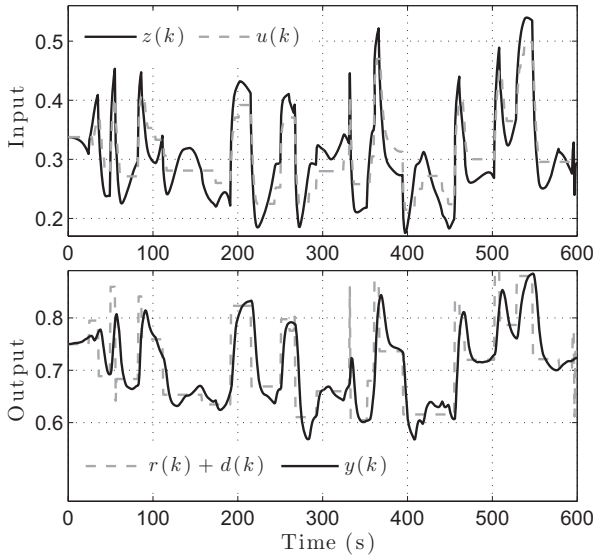


Fig. 7. Input-output data and external test signal of the high disturbance simulation.

The friction and process model parameters are estimated using 600 samples and a 1s sampling period. From (18) one has $\max(S) = 35\%$. A set of candidate values of the pair (S,J) is generated with a resolution of 1% and the process model was estimated using: $m = p = 3$, $r = 35$, $n_c = 0$, $n_d = 6$ and $l = 3$.

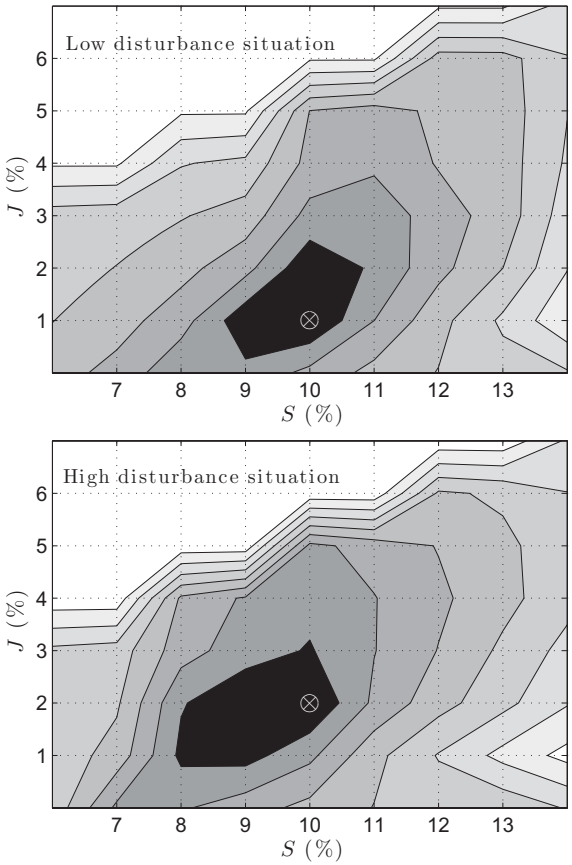


Fig. 8. Level curves of the prediction error $\mathcal{C}$.

The behavior of the prediction error $\mathcal{C}$, in both disturbance situations, is shown in Figure 8. Darker locations indicate lower prediction errors and the symbol "$\otimes$" indicates the minimum. In both disturbance situations, the parameter S is exactly estimated.

On the other hand, $\hat{J} = 1\%$ and 2% for the low and high disturbance scenarios, respectively. The slight misfit obtained in lower disturbance situation has two reasons: (1) the influence of the parameter S is prominent if compared to J and (2) the occurences of slip-jumps in the low disturbance simulation is minor (21 slip-jumps against 26 provided by the major disturbance simulation).

The Nyquist plot of linear block estimate from both disturbances scenarios are compared to the actual one in order to check if the process dynamics were incorporated. From Figure 9 it can be seen that the estimation from the low disturbance dataset provided better results. Therefore, it is clear that the disturbances degrade the accuracy of the identified linear dynamic block.

To get a better insight about frequency domain errors, the Bode plot of the estimates is depicted in Figure 10. Although the Nyquist curves suggest that the dynamic block estimate from the high disturbance simulation data presents a larger misfit, one can see that the errors related to the actual frequency response are acceptable for practical purposes.

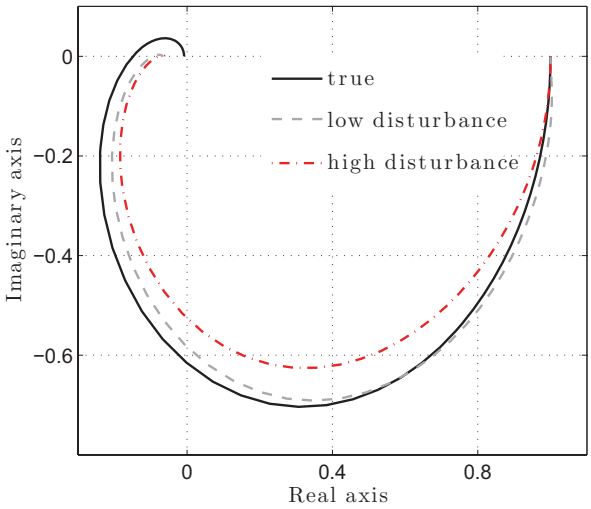


Fig. 9. Nyquist plot of $\hat{G}(q)$ obtained from both disturbance scenarios.

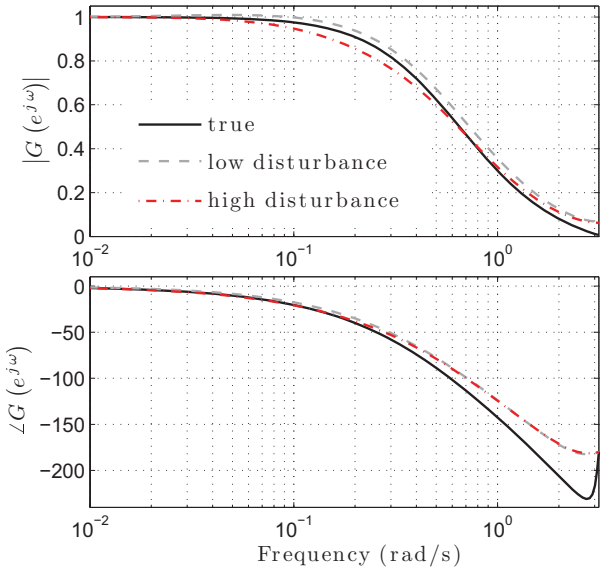


Fig. 10. Bode plot of the estimated dynamic models.

Finally, the nonlinear block fit is investigated. The true process nonlinearity and the estimates from low and high disturbance scenarios, denoted as $\hat{f}_{(low)}(\cdot)$ and $\hat{f}_{(high)}(\cdot)$, are presented in Figure 11. Despite adopting a parameterization different from the cubic splines during the simulations, the process nonlinearity is accurately estimated in both disturbance conditions.

Comparing the results achieved in each of the disturbance situations, one can see that the process steady state curve fit is better when the disturbance level is lower. Nevertheless, the estimation performance deterioration is slight, in spite of the substantial increase (almost 10 times higher) in the disturbance level.

6. CONCLUSIONS

The results provided by the simulated example suggests that the proposed procedure that jontly identifies the

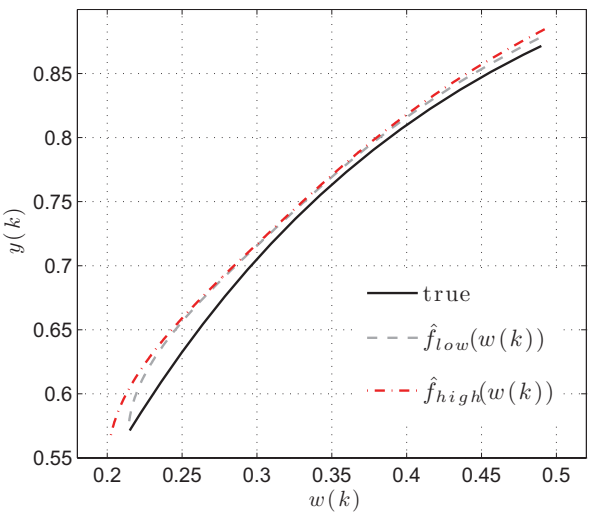


Fig. 11. Actual and estimated process nonlinearity.

friction and the nonlinear process model parameters is promising.

Moreover, the estimation results indicate that the GMN test signal is suitable for the process model identification. On the other hand, this excitation can not be adjusted in order to control the ocurrences of slip jumps. Thus, the GMN is insufficient to guarantee accurate estimates of the parameter J. An alternative to deal with this drawback is to combine a multi-level randon noise with a staircase excitation.

Another aspect that shoud be emphasized is that, when the process nonlinearity is severe, the Wiener model can be used to develop nonlinear controllers, such as gain schedule strategies.

Results of the procedure proposed here applied to industrial data is under development.

ACKNOWLEDGEMENTS

The authors thank CAPES for the Dr. scholarship granted to Rodrigo Alvite Romano.

REFERENCES

M. A. A. S. Choudhury, S. L. Shah, and N. F. Thornhill. Detection and quantification of control valve stiction. In *IFAC Symposium on dynamics and control of process systems (DYCOPS)*, Cambridge, MA, USA, July 5-7 2004.

M. A. A. S. Choudhury, M. Jain, S. L. Shah, and D. S. Shook. Stiction - definition, modelling, detection and quantification. *Journal of Process Control*, 18:232–243, 2008.

L. Desborough and R. Miller. Increasing customer value of industrial control performance monitoring - Honeywell's experience. In *6th International Conference on Chemical Process, CPC - VI*, pages 169–189, Tucson, USA, 2001.

U. Forssell. *Closed-loop Identification: Methods, Theory, and Applications*. Licentiate thesis no. 566, Department of Electrical Engineering, Linköping University, Linköping, Sweden, Mar 1999.

C. Garcia. Comparison of friction models applied to a control valve. *Control Engineering Practice*, 16(10): 1231–1243, 2008.

M. Kano, M. Hiroshi, H. Kugemoto, and K.Shimizu. Practical model and detection algorithm for valve stiction. In *IFAC Symposium on dynamics and control of process systems (DYCOPS)*, Cambridge, MA, USA, July 5-7 2004.

P. Lancaster and K. Šalkauskas. *Curve and Surface Fitting: An introduction*. Academic Press, London, 1986.

L. Ljung. *System Identification: theory for the user*. Prentice Hall, New Jersey, 2nd edition, 1999.

R. Srinivasan, R. Rengaswamy, S. Narasimhan, and R. Miller. Control loop performance assessment. 2. Hammerstein model approach for stiction diagnosis. *Industrial & Engineering Chemistry Research*, 44(17): 6719–6728, 2005.

Y. Zhu. *Multivariable System Identification for Process Control*. Elsevier Science, Oxford, 2001.

Y. Zhu. Distillation column identification for control using Wiener model. In *1999 American Control Conference*, Hyatt Regency San Diego, California, USA, 1999.

Control Loop Performance Monitoring using the Permutation Entropy of Error Residuals

Rachid A. Ghraizi, Ernesto C. Martínez*, César de Prada****

**University of Vallodolid, Department of Systems Engineering and Automatic Control, Valladolid 47011, Spain (e-mails:{rachidag, prada}@autom.uva.es)*
***National Scientific Research Council, INGAR(CONICET), Avellaneda 3657, Santa Fe, S3002 GJC, Argentina (Tel: +54 342 4534451; e-mail: ecmarti@ santafe-conicet.gob.ar)*

Abstract: The predictability of a control-loop behavior beyond its control horizon is an unambiguous indication of loop malfunctioning. Based on the dynamic complexity of the error residual time series the permutation entropy is proposed to define a sensitive index for performance monitoring using data from close-loop operation. A generic framework to understand and quantify the distinctive increase in predictability of the controller error resulting from ill-tuning, sensor errors and actuator faults using a entropy-like index is presented. The dynamic complexity of a well-performing control loop should correspond to the maximum entropy. As loop performance degrades the entropy of its residual time series decreases and any loss of dynamic complexity in the control system gives rise to an increase of the predictability of the control error time series. Results obtained using the proposed performance index along with its confidence interval for industrial data sets are presented to discuss the influence of the sample size, control horizon, and variance estimation in the assessment of close-loop performance.

Keywords: Control-loop performance; Monitoring; Permutation entropy; Fault detection and diagnosis; Error predictability; Ordinal patterns; Time-series analysis.

1. INTRODUCTION

Control loops implementing a hierarchy of functions for process regulation and optimization are the cornerstone of safety and economy in process plants (Thornhill et al., 1999). Many loops are just PID controllers whilst other may be more advanced ones, such as inferential loops, MPCs and real-time optimizers working on top of the regulation layer. It is well known that in most industrial environments the behavior of control loops deteriorate with time due to a number of reasons, e.g. plant-wide perturbations, fouling, utility constraints and raw material variability. Accordingly, process dynamic characteristics change over time and, if not properly maintained, most control loops will perform poorly after some time, which can lead to degraded process operation. In particular, ill-functioning of the regulation layer can easily cancel the benefits of advanced control systems and real-time optimization (Jelali, 2006; AlGhazzawi and Lennox, 2009). With the increasing complexity of control structures and the sheer number of controllers in modern process plants, the automation of performance monitoring tasks is mandatory.

Systematic assessment of SISO control loops can be traced back to the seminal work of Harris (1989) who related the performance of a single-loop control system to the controller errors of a minimum variance controller. The latter, even though it is rather impractical to be implemented, serves as a performance benchmark to provide a lower bound for the variance of the controlled variable. On this basis, the well-known Harris index is defined as the ratio of the variance achievable using minimum variance controller to the variance measured under the current control law (Desborough and Harris, 1992, 1993). As the value of this statistic is reduced then so too does the measured performance of the control system. The key advantage of the Harris approach to control loop monitoring is that only routine close-loop operating data are required to determine the performance of the control system. This fact has made the approach very attractive to industry and it is now applied as a matter of routine by many companies. However, a disadvantage of the Harris index is that it is based on a rather extreme (in terms of cost and energy involved) behaviour and no hints are provided for characterizing the behaviour a well-performing realistic controller based on the control task for which it was designed. Also, it is difficult to pinpoint an informative threshold for the Harris index to differentiate between normal and faulty operation of a control loop.

Based on the insightful concept of control horizon, Thornhill *et al.* (1999) proposed the predictability of the error time series to characterize the performance of a SISO controller. The predictability of a control-loop behaviour beyond its control horizon is an unambiguous indication of loop malfunctioning in biological systems (Li, Ouyang and Richards, 2007). Along this research avenue, Ghraizi *et al*. (2007), proposed a practical index for performance monitoring of a control loop based on the analysis of the predictability of the error time series and emphasizes proper

selection of the control horizon using engineering judgment and the amplitude and frequency of disturbances to which the loop is designed for. To develop ideas further, Martínez and de Prada (2007) resort to ordinal analysis methods of the error time series to define a performance index for performance monitoring based on the permutation entropy.

In this work, the interplay between predictability of controller behaviour and its dynamic complexity for performance monitoring is highlighted by resorting to the residual error time series, which is obtained using a regression model. A generic framework to understand and quantify the distinctive increase in predictability of the controller error resulting from ill-tuning, sensor errors and actuator faults using an entropy-like index is proposed. A well-performing controller should behave so that the sequence of residuals in the error series looks like one generated using *i.i.d.* samples from a random walk and the corresponding dynamic complexity is thus maximum. Accordingly, ordinal patterns in the error residuals will all be equally probable and the corresponding permutation entropy will be then the highest possible.

2. MONITORING METHODOLOGY

2.1 *Predictability analysis*

The performance-monitoring concept revolves around the idea of predictability of controller behaviour beyond a chosen horizon *b*. If a control loop exhibits "good" performance, it should be able to cancel any disturbance entering the loop up to present time *t*, or follow a set point change correctly, after some sensible time interval *b* (expressed in terms of sampling periods). Then, it can be argue that, as from time *t*+*b* onwards, the error time series cannot be distinguished from a random walk stochastic process so that it cannot be predicted at all using information up to time instant *t* (see Fig. 1 for details). Nevertheless, over the control horizon *b*, the controller behaviour is fully predictable since it corresponds to its own control policy built-in by design. By contrast, error time series of a control loop exhibiting "poor" performance, will show patterns of behaviour (oscillations, steady-state errors, etc.) which can be predicted after time instant *t*+*b* using present and past measurements.

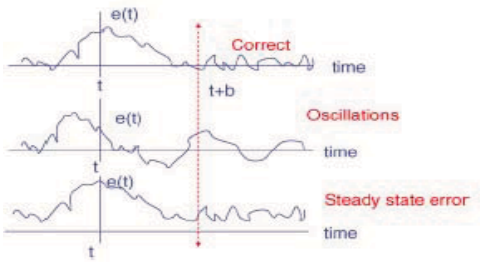


Fig. 1. Error patterns and their predictability

The most sensitive approach to detect patterns of predictability in the time series is analyzing the time series of error residuals $r(t)$ which are obtained using an inductive model to predict future errors.

Let's denote by $e(t)$ the controller error defined as

$$e(t) = \omega(t) - y(t) \tag{1}$$

Where $\omega(t)$ stands for the desired set-point at any time t, and $\hat{e}(t)$ stands for the prediction of such error based on past values of the controller error. The difference between the actual and predicted controller errors is the residue *r(t)* whose time series has a dynamic complexity closely related to the predictability patterns in the controller error time series

$$r(t) = e(t) - \hat{e}(t) \tag{2}$$

The error prediction $\hat{e}(t)$ can be obtained in different ways, but the easiest alternative is using the regression model

$$\begin{aligned}\hat{e}(t+b) = e_o + e_1(t) + e_2(t-2) + e_3(t-3)....\\ + e_m(t-m+1)\end{aligned} \tag{3}$$

Where time indices refer to sampling periods, *m* is the model order and a_i is the unknown parameters, which are fitted using a dataset of size *n* by means of the least-square regression:

$$[a_0, a_1, ..., a_m]^T = (X^T X)^{-1} X^T Y \tag{4}$$

Where

$$X = \begin{bmatrix} 1 & e(1) & e(2) & \cdots & e(m) \\ 1 & e(2) & e(3) & \dots & e(m+1) \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ 1 & e(n-b-m+1) & \cdots & \cdots & e(n-b) \end{bmatrix} \tag{5}$$

And

$$Y = \left[e(m+b)\ e(m+b+1) \cdots e(n)\right]^T \tag{6}$$

It is worth noting that for a well-performing controller in a given time interval the sequence of error residuals is a chaotic, completely random, and non-stationary stochastic process exhibiting maximum dynamic complexity. The reader is referred to the work of Peng et al. (2009) for an interesting discussion on the meaning of regularity and dynamic complexity in physiologic time series from highly controlled biological systems. To quantify the dynamic complexity of residuals there are several options.

In a previous work, the authors, Ghraizi *et al*. (2007), used a performance index based on the ratio between the variance of the residuals and the variance of the errors:

$$PI = \frac{\sigma_r^2}{\sigma_e^2} \tag{7}$$

Assuming that in a perfectly predictable loop the variance of

the residual would be zero, while non-predictable random walk would give a variance similar to the loop error, this expression would provide an index ranging from zero to one that will measure the performance of the controller. In order to obtain a confidence interval of the index the following analysis can be performed:
It is known that the following ratio

$$\frac{(n-1)\hat{\sigma}^2}{\sigma^2} \tag{8}$$

between the estimated and real variance for an stochastic process must follow a χ^2 distribution with n-1 degrees of freedom. Applying this line of reasoning to the residuals and the controller errors and dividing them, we can obtain:

$$\frac{\sigma_r^2\hat{\sigma}_e^2}{\sigma_e^2\hat{\sigma}_r^2} \tag{9}$$

Which will follow a F-distribution with n-1,n-1 degrees of freedom. Hence:

$$\begin{aligned} &P(F_{1-0.5\alpha,n-1} \le \frac{\hat{\sigma}_e^2\sigma_r^2}{\sigma_e^2\hat{\sigma}_r^2} \le F_{0.5\alpha,n-1}) = 1-\alpha \\ &P(\frac{\hat{\sigma}_r^2}{\hat{\sigma}_e^2}F_{0.5\alpha,n-1}^{-1} \le \frac{\sigma_r^2}{\sigma_e^2} \le \frac{\hat{\sigma}_r^2}{\hat{\sigma}_e^2}F_{0.5\alpha,n-1}) = 1-\alpha \end{aligned} \tag{10}$$

Will can be used to compute the 100(1-α) % interval of confidence for the *PI* index defined in (7).

In practice, when this index is computed, large confidence interval appears sometimes, mainly when loop performance degrades, which reduces the interest in the above method. An alternative not based on statistical assumptions, which are always difficult to verify, would be desirable. In this regard, an appealing and sound choice is resorting to an entropy-like index based on ordinal patterns of the residual time series.

2.2 *Residual order patterns*

The complexity of a residual time series can be quantified by means of its symbolic dynamics. A new permutation method was proposed by (Bandt and Pompe, 2002; Bandt, 2005) to map a continuous time series onto a symbolic sequence; the statistics descriptive of the dynamic complexity of the symbolic time series is called permutation entropy. Given a data set for the scalar residual time series $r(t), t = 1,....,n$, the local order of the series can be characterized by patterns in vectors $\Re(t)$ ensembled as follows

$$\Re(t) = [r(t), r(t+\ell),...r(t+(\kappa-1)\ell))] \tag{11}$$

where κ is the embedded dimension parameter and ℓ is the lag parameter (here $\ell = 1$). Then entries in each $\Re(t)$ are arranged in increasing order which allows assigning to it one out of the possible order patterns. For κ different numbers, there will be $\kappa!$ possible order patterns π, which are also called permutations. In Fig. 2 the six order patterns for $\kappa = 3$ are shown. Let $f(\pi)$ denote the frequency of permutation π in the data set whereas $\rho(\pi) = f(\pi)/(n-(\kappa-1)\ell)$ is the relative frequency. For a perfectly working controller the relative frequencies should all be close to $1/\kappa!$

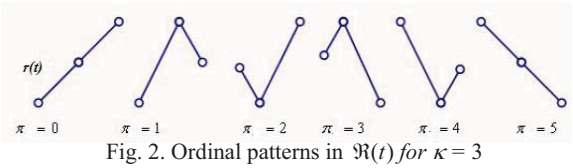


Fig. 2. Ordinal patterns in $\Re(t)$ *for* $\kappa = 3$

2.3 *Permutation entropy and performance monitoring*

The local permutation entropy of order κ for the error residual time series is defined as

$$H_\kappa = -\sum_{\pi=1}^{\kappa!} \rho(\pi)\ln\rho(\pi) \tag{12}$$

The largest possible value for the permutation entropy will correspond to the perfectly working controller where all $\kappa!$ permutations are equally probable which coincides with a residual time series of maximum complexity where the permutation entropy is $\ln \kappa!$

Permutation entropy depends on the selection of κ. When κ is too small a value (say less than 3), the scheme will not work, since there are only very few distinct states for characterizing the control system behavior. For too large values of κ (greater than 6), the number $\kappa!$ of permutations which can appear in the time series can result in computer memory problems, due to the large number of data points that need to be examined. In the present work, only values of $\kappa = 3$, 4 or 5 will be used.

For loop monitoring, the permutation entropy of the residual time series is obtained from a sample where the total number of patterns counted is n and the tally number for the *ith* pattern in the sample is denoted by f_i

$$\hat{H}_\kappa^n = -\sum_{i=1}^{\kappa!}\left(\frac{f_i}{n}\ln\frac{f_i}{n}\right) \tag{13}$$

The corresponding variance for this sample estimation of the permutation entropy is (see Moddemeijer, 1989, for details)

$$Var\left(\hat{H}_\kappa^n\right) = \frac{1}{n}\left(\sum_{i=1}^{\kappa!}\frac{f_i}{n}\ln^2\frac{f_i}{n} - \left(\frac{f_i}{n}\ln\frac{f_i}{n}\right)^2\right) \tag{14}$$

Based on equations (13) and (14) and a sample of size n, the following performance index is proposed

$$PI = \frac{\hat{H}_\kappa^n}{\ln\kappa!}, \kappa = 3,4,... \tag{15}$$

Since $\ln\kappa!$ is a constant, the variance for the sample estimation of the performance index can be written as

$$Var(PI) = \frac{Var\left(\hat{H}_\kappa^n\right)}{\ln\kappa!}, \kappa = 3,4,... \tag{16}$$

The highest value of *PI* is one, which means the error residual time series is complete random and is its dynamics is very complex; the smallest possible value of *PI* is zero, which

means the error residual time series is highly regular.

Eq. (14) is very important for the following reasons: it is possible to make a very reliable characterization of the variance of a sample-based estimation of the performance index *PI* in Eq. (15). For a well-performing control loop, since the probability for ordinal patterns are all the same, after elementary algebra steps in Eq. (14) an *exact* measure of the variance for the performance index is obtained

$$\frac{\left[\ln\left(\frac{1}{\kappa!}\right)\right]^2(\kappa!-1)}{\ln\kappa!\; n\;\kappa!} \qquad (17)$$

As can be readily calculated, this variance for a properly working loop is very small even for small sample sizes. For example, for n=1000 and κ=3, the variance of *PI* is ≅0.00149 for a perfectly working controller. For a much smaller sample size such n=100 is still rather small (≅0.0149).

To compute a 100(1-α) % confidence interval for the *PI* index estimated through Eq. (15), the Student's *t*-distribution is assumed so that the sample variance in Eq. (16) is used to define upper/lower limits in the usual way as follows

$$\pm t_{1-\alpha/2,n-1}\sqrt{\frac{Var(PI)}{n}} \qquad (18)$$

where α defines the chosen level of confidence.

3. RESULTS

In order to show the applicability of the proposed method, several data sets from an industrial site have been considered. They correspond to routine plant data of a set of typical control loops for pressure, flow, etc. The first one is displayed in Fig.3 and contains 9000 samples of the controlled and manipulated variables as well as the set point of a pressure control loop.

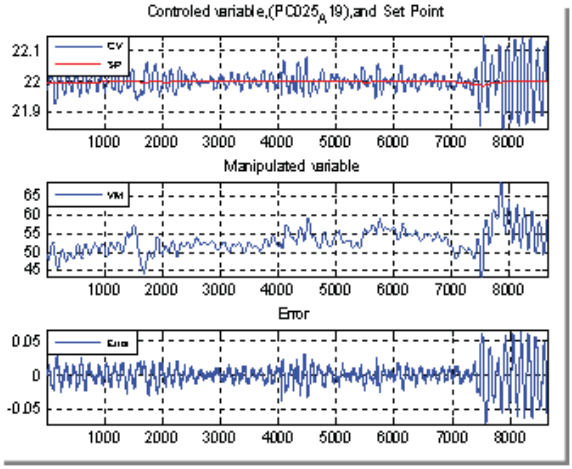


Fig.3 Data from a pressure control loop. Upper graph: Set point (in red) and gauge pressure. Middle graph: control valve signal. Bottom graph: error between set point and pressure readings.

Despite the fact that pressure readings stand close to the set point, the loop experiments an oscillatory behavior, likely as result of a too tight tuning. This can be better observed in a close-up to the data, as the one shown in Fig.4.

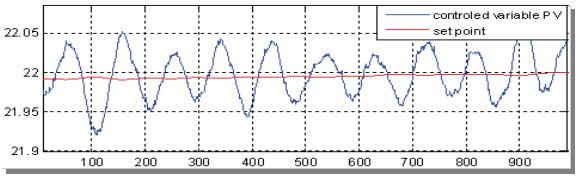


Fig.4. A detailed view of the first 1000 data of the time evolution of pressure and its set point for the example data set shown in Fig.3.

Results obtained from the application of the proposed method to pressure loop are discussed next. Fig.5. displays the actual error and its predictions computed from the expression (3). As can be seen, predictability is significantly high, as one could expect in a badly tuned control loop whereas residuals are small. The residual time series is displayed in the upper part of Fig. 6, exhibiting a certain regularity, whereas in the bottom part the values of the proposed performance index (15) computed at regular time intervals every 120 data are shown for the case κ = 3 and b = 12.

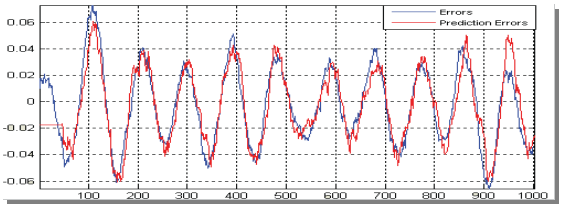


Fig.5. A detailed view of the errors and its predictions (in red) for the first 1000 data using prediction formula (3).

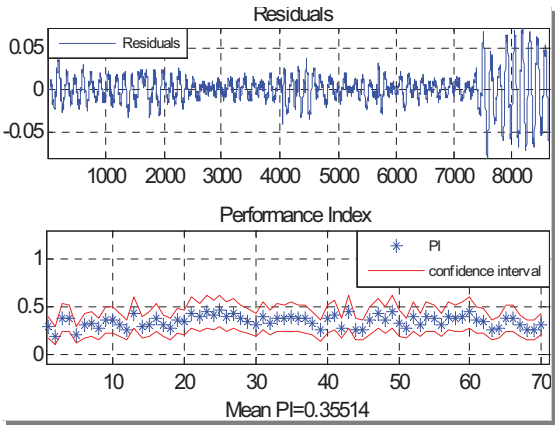


Fig.6. Upper graph: Residuals of the predictions computed with expression (2) for the example of Fig.3. Lower graph: Performance index computed from (16).

As can be seen, in this example, sample estimation the performance index have an average value of 0.35, which is an indication of poor performance of the pressure loop. For the sake of comparison, the *PI* computed from (7) is displayed in Fig.7. Even though consistent results are obtained, but the

confident limits rise up to 1 which creates a great deal of uncertainty in the estimation of this performance index.

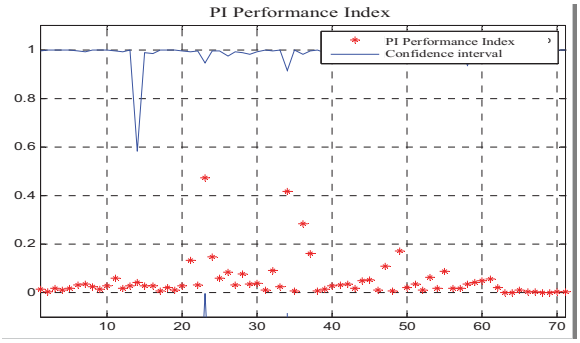


Fig.7. Performance index (in red) computing with expression (7) for the example of Fig.3. Blue line gives the upper confidence limit based on Eqs. (9) and (10).

Data from a second case study are given in Fig. 8. This time data correspond to a flow control loop that is the slave in a cascade configuration. 16,000 data poins are collected and, as can be seen, the loop behaviour is quite good, with fine set point tracking and moderate control signal changes, except in the range for data poins from 12,000 to 13,000, where extreme values of the set point from the master loop lead to saturation of the manipulated variable.

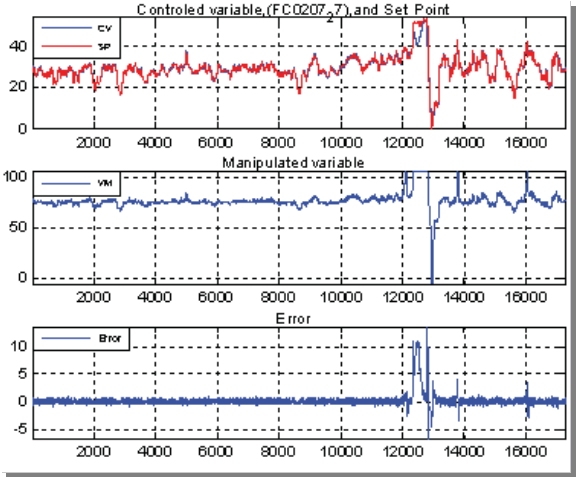


Fig.8. Data from a slave flow control loop. Upper graph: Set point (in red) and flow readins. Middle graph: control valve signal. Bottom graph: error between set point and actual flow.

As can be seen in the close-up displayed in Fig.9, there is a good following of the set point and the predictions of the errors computed with the expression (3) differ from the actual errors, so that the residuals (2), displayed in Fig.10, approach to a random walk, as expected in a loop with good behaviour.

The performance index (15) has been computed using the values $\kappa = 3$ and $b = 12$, at regular time intervals that include 120 data points. The numerical values displayed on the bottom of Fig.10 give regular values around 0.82 that is a reasonable value for this entropy-like index when all patterns are almost equiprobable.

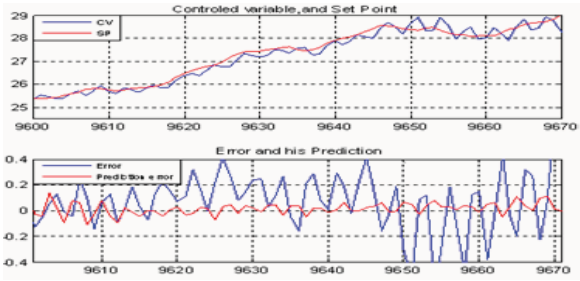


Fig.9. Upper graph: A close-up of data of the time evolution of flow and its set point (in red) for the example of Fig.8. Lower graph: A detailed view of the errors and its predictions (in red) for this range of data using prediction formula (3).

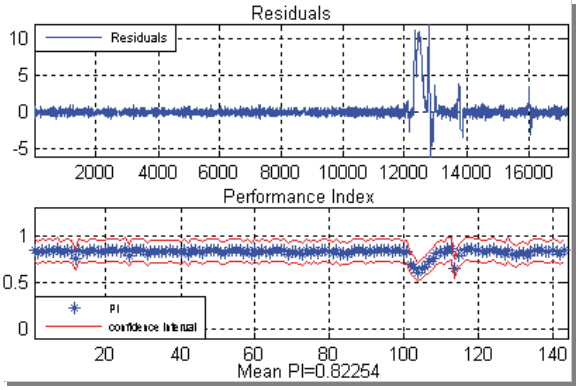


Fig.10. Upper graph: Residuals of the predictions computed with expression (2) for the example of Fig.8. Lower graph: Performance index computed from (16).

Nevertheless, in a set of intervals from data 12,000 on, where the manipulated variable is saturated, the predictions of the errors can be made very well, as can be observed in Fig.11, where the errors and its predictions computed with formula (3) are given.

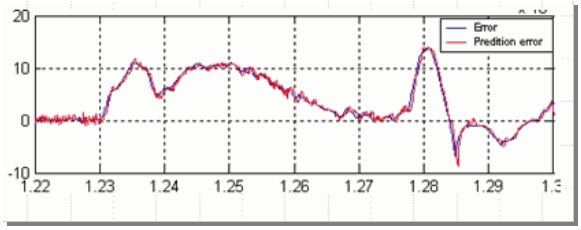


Fig.11. A detailed view of the errors and its predictions (in red) for a range of data above sample 12,000 of the example of Fig.8, using prediction formula (3).

In this case, the residuals are small and do not follow a random walk stochastic process, as can be seen in the corresponding range of variation shown in Fig.10. Accordingly, the *PI* decreases, indicating performance degradation of the loop in this portion of the data set.

For comparison, the *PI* computed with expression (7) and blocks of about 1000 data is given in Fig.12. As we can see, the index is between 0.75 and 0.9 giving consistent indications about the goodness of the loop behaviour, but at

the time intervals where the saturation occurs it drops to 0.1, 0.3 as expected. Nevertheless, the confidence bands increases at this precise times, decreasing the certitude of the diagnosis.

A final test was made to compare previous results with the ones provided by the Harris index for this case study, which are displayed in Fig.13. The values for the Harris index range from 0.3 to 0.5 for most samples, indicating that a margin for improvement exists in relation to the best possible linear controller -the minimum variance one- but it is worth noting that this index does not provide a direct measurement of the loop performance. Notice also that the index drops to 0 and 0.1 in the critical range when the saturation of the manipulated variable occurs.

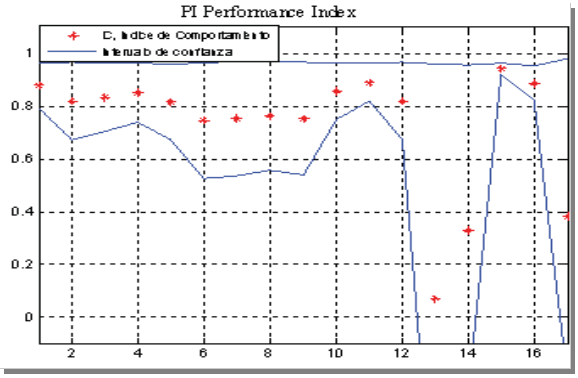


Fig. 12. Performance index (in red) computed with expression (7) for the example of Fig.8. Blue line gives the upper confidence limit.

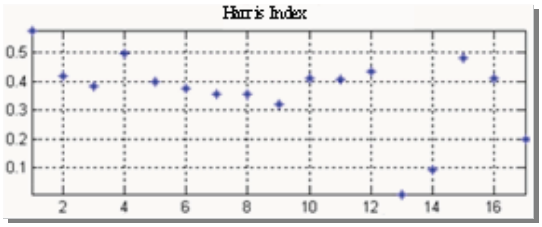


Fig. 13. Sample estimation of the Harris index for the example in Fig. 8 computed from batches of 1,000 data points.

4. CONCLUSIONS

A new index for performance assessment of control loops using normal operating plant data has been proposed. It combines the idea of predictability of the controller error at a point in time beyond the desired settling time of the loop, with an analysis of the corresponding sequence of prediction residuals based on ordinal methods along with the concept of local permutation entropy. The main advantage of the Performance Index defined in this way is the fact that no statistical assumptions are made on the residuals, which allows for a crisp interpretation of sample estimation of the performance index. Moreover, the entropy-like index is easy to compute and can be applied to single isolated loops as well as to cascades or other control configurations including model predictive controllers. For industrial data sets, the proposed *PI* has provides consistent results.

To highlight several advantages some comparisons were made with other indices such the Harris index and a previous *PI* based on the error predictability idea. The proposed method based on the permutation entropy can be applied in real time with minimum computational costs which opens the possibility of automatic supervision of hundreds of control loops of a typical process plant. Also, linking information content with predictability of error residuals is a novel idea for loop monitoring using dynamic complexity.

REFERENCES

AlGhazzawi, A. and Lennox, B. (2008). Model predictive control monitoring using multivariate statistics. *J. of Process Control* (in press).

Bandt, C. (2005). Ordinal time series analysis. *Ecological Modelling* 182, 229–238.

Bandt, C. and B. Pompe (2002). Permutation Entropy: A Natural Complexity Measure for Time Series. *Physical Review Letters* 88, 174102.

Desborough, L., & Harris, T. (1992). Performance assessment measures for univariate feedback control. *Canadian J. of Chemical Engineering* 70, 1186–1197.

Desborough, L., & Harris, T. (1993). Performance assessment measures for univariate feedforward/feedback control. *Canadian J. of Chemical Engineering* 71, 605–616.

Ghraizi, R. A. *et al.* (2007). Performance monitoring of industrial controllers based on the predictability of controller behavior. *Computers and Chemical Engineering* 31, 477-486.

Harris, T. J. (1989). Assessment of control loop performance. *Canadian J. Chemical Engineering*, 67, 856–861.

Jelali, M. (2006). An overview of control performance assessment technology and industrial applications. *Control Engineering Practice,* 14(5).

Li, X., Ouyang, G. and Richards, D. A. (2007). Predictability analysis of absence seizures with permutation entropy. *Epilepsy Research* 77, 70-74.

Martínez, E. C. and de Prada, C. (2007). Control Loop Performance Assessment using Ordinal Time Series Analysis. *Computer-aided Chemical Engineering* 24, 261-266.

Moddemeijer, R. (1989). On the estimation of the entropy and mutual information of continuous distribution. *Signal Processing* 16, 233-248.

Peng, C. K., Costa, M. and Goldberger, A. L. (2009). Adaptive data analysis of complex fluctuations in physiologic time series. *Advances in Adaptive Data Analysis* 1(1), 61-70.

Thornhill, N. F., Oettinger, M., and Fedenczuk, P. (1999). Refinery-wide control loop performance assessment. *J. of Process Control* 9, 109–124.

Performance Assessment of Decentralized Controllers

Antonius Yudi Sendjaja * **Vinay Kariwala** *

* *Division of Chemical and Biomolecular Engineering, Nanyang Technological University, Singapore 637459 (e-mails: ayudi@pmail.ntu.edu.sg, vinay@ntu.edu.sg)*

Abstract: Minimum variance (MV) benchmark is useful for identifying variance reduction opportunities in industrial control systems. During the past two decades, MV benchmarks for single-input single-output (SISO) and multi-input multi-output (MIMO) systems have been proposed. These MV benchmarks do not account for the structure of the decentralized or multi-loop controllers, which are used almost exclusively for regulation purposes in process industries. Due to this drawback, the available MV benchmarks can lead to incorrect conclusions regarding the performance of decentralized controllers. This paper aims to fill this gap. For performance assessment of decentralized controllers on a loop-by-loop basis, we present a simple modification of the available MV benchmark for SISO systems. For simultaneous performance assessment of all loops, we present a method for computing a tight lower bound on the achievable output variance. In the latter approach, the non-convexity of the resulting optimization problem is handled using sums of squares programming. The usefulness of the proposed benchmarks is evaluated using examples drawn from the literature.

Keywords: Decentralized control, Minimum variance control, Performance limits, Performance monitoring, Sums of squares programming.

1. INTRODUCTION

The performance of a well-designed control system can degrade over time due to changes in operating conditions and disturbance dynamics. Controller performance assessment is useful for identifying the opportunities for performance improvement of industrial controllers. Among the various available methods (Qin, 1998; Jelali, 2006), minimum variance (MV) benchmarking is one of the most promising methods for controller performance assessment. In this approach, the controller is deemed to provide satisfactory performance, if MV benchmark (ratio of least achievable and observed output variances) is close to 1. On the other hand, reduction in output variance is considered to be feasible through controller retuning, when the MV benchmark is significantly lower than 1.

The origin of MV benchmark can be traced back to Åström (1970), who demonstrated that the achievable output variance for a single-input single-output (SISO) process under feedback control depends on the first few impulse response coefficients of the disturbance model. Harris (1989) showed that with *a priori* knowledge of time delay, MV benchmark can be estimated using closed loop operating data and established it as a tool for performance assessment of SISO systems. Using the concept of interactor matrices, Harris et al. (1996) and Huang et al. (1997) proposed MV benchmark for multi-input multi-output (MIMO) systems.

This paper focusses on performance assessment of decentralized or multi-loop controllers, which are used almost exclusively for regulation purposes in process industries. Though useful, the available MV benchmarks for SISO and MIMO processes show limitations, when applied to processes under decentralized control. The conventional approaches for performance assessment of decentralized controllers include (see *e.g.* (Harris et al., 1996) and (Huang et al., 1997) for examples):

(1) *Loop by loop analysis*: The performance of individual loops is assessed independent of each other using MV benchmark for SISO processes.
(2) *Simultaneous analysis*: The performance of all loops is assessed simultaneously using MV benchmark for MIMO processes.

The MV benchmark for SISO processes assumes that the loop under consideration is being operated in isolation from the rest of the process and thus inherently views the process as being diagonal; see Figure 1. Due to this assumption, it may be possible to improve the performance of the existing controller further than indicated by the MV benchmark; see Section 3 for details. On the other hand, MV benchmark for MIMO processes ignores the diagonal structure of the decentralized controller and thus has more degrees of freedom for variance minimization than are available in the actual controller. Using a simple 2×2 process, we demonstrate in Section 4 that the least output variance that can be achieved using a diagonal controller can be four times higher than that can be achieved using a full multivariable controller. In summary, ignoring the controller structure can lead to incorrect conclusions regarding performance assessment of decentralized controllers.

The derivation of an approach for MV benchmarking of decentralized controllers requires characterization of the

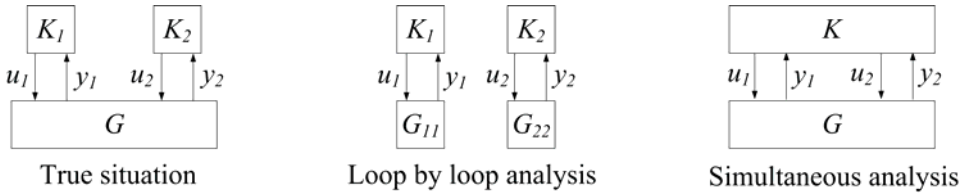


Fig. 1. Insufficiency of available MV benchmarks for performance assessment of decentralized controllers

least achievable output variance and its subsequent estimation from closed-loop data. This paper mainly focusses on the first issue. We first propose an MV benchmark for loop-by-loop analysis, where the presence of other loops is accounted for. The proposed result requires a small modification of the existing MV benchmark for SISO processes. It is further shown that the modified MV benchmark for loop-by-loop analysis can be directly estimated from closed-loop data with *a priori* knowledge of the delays of the different elements of the process model. An interesting insight is that for processes under decentralized control, the pre-whitening of output data using algorithms such as filtering and correlation (FCOR) algorithm (Huang and Shah, 1999) does not necessarily provide the first few impulse response coefficients of the disturbance model, as is traditionally believed.

The derivation of MV benchmark for simultaneous analysis of decentralized controller is more challenging. This happens as the optimization problem involving minimization of output variance becomes non-convex, once the diagonal structure is imposed on the controller (Sourlas and Manousiouthakis, 1995; Rotkowitz and Lall, 2006). With this difficulty, one may alternately look for tight upper and lower bounds on the least achievable output variance using decentralized controllers. Clearly, any suboptimal tuning strategy for the decentralized controller provides an upper bound on the least achievable output variance. Some approaches for finding upper bounds on least achievable output variance have been reported using non-convex optimization (Ko and Edgar, 1998; Jain and Lakshminarayanan, 2007) or by utilizing the structure of the optimization problem (Yuz and Goodwin, 2003; Kariwala et al., 2005). Recently, Kariwala (2007) addressed the more difficult problem involving derivation of a tight lower bound on the least achievable output variance, where an explicit bound is proposed by considering those impulse response coefficients of the closed-loop transfer function between disturbances and outputs, which depend linearly on the controller parameters. In general, however, the lower bound proposed in (Kariwala, 2007) can be conservative due to the neglected impulse response coefficients.

In this paper, we show that though nonlinear, the impulse response coefficients of the closed-loop transfer function between disturbances and outputs can be represented as polynomials in unknown controller parameters. Subsequently, the non-convex optimization problem related to the minimization of output variance is solved using sums of squares (SOS) programming (Parillo, 2000). This result is further extended to find a lower bound on the least achievable output variance, when the individual subcontrollers of the decentralized controller are restricted to be of proportional-integral-derivative (PID) type. The estimation of these lower bounds is difficult from closed-loop data only and the knowledge of process model is required. Nevertheless, the derivation of lower bound on the least achievable output variance can itself be seen as a major step towards systematic performance assessment of decentralized controllers.

2. PROBLEM FORMULATION

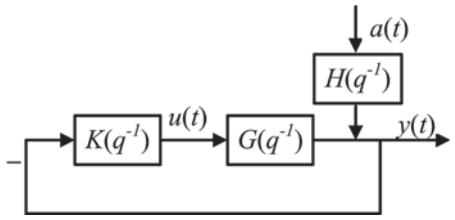


Fig. 2. Block diagram of closed-loop system

We consider the closed-loop system shown in Figure 2. For this system, we denote $G(q^{-1})$ and $H(q^{-1})$ as the process and disturbance models, respectively, such that

$$y(t) = G(q^{-1})\,u(t) + H(q^{-1})\,a(t) \tag{1}$$

Here, $y(t), u(t)$ and $a(t)$ are controlled outputs, manipulated variables and disturbances, respectively. We make the following simplifying assumptions:

(1) $G(q^{-1})$ and $H(q^{-1})$ are stable, causal transfer matrices, contain no zeros on or outside the unit circle except at infinity (due to time delays), and are square having dimensions $n_y \times n_y$.
(2) $a(t)$ is a random noise sequence with unit variance.

When $H(q^{-1})$ contains zeros outside the unit circle, these zeros can be factored through an all pass factor without affecting the noise spectrum (Huang and Shah, 1999). Further, there is no loss of generality in assuming that the system is affected by noise having unit variance, as the disturbance model can always be scaled to satisfy this assumption. For notational simplicity, we drop the arguments q^{-1} and t in the subsequent discussion.

Our objective is to find the least achievable value of $\mathrm{Var}(y)$ with respect to the controller K, *i.e.*

$$J_{\mathrm{decen}} = \min_{K} \mathrm{Var}(y) = \min_{K} E[\mathrm{tr}(y\,y^T)] \tag{2}$$

where K is assumed to have a diagonal structure, *i.e.* $K = \mathrm{diag}(K_1, K_2, \cdots, K_{n_y})$. In (2), $E(\cdot)$ and $\mathrm{tr}(\cdot)$ denote the expectation and trace operators, respectively. The pairings are considered to be selected on the diagonal elements of G.

Based on (2), the MV benchmark for decentralized controller can be defined as

$$\eta_{\text{decen}} = \frac{J_{\text{decen}}}{E[\text{tr}(y\, y^T)]} \tag{3}$$

where $E[\text{tr}(y\, y^T)]$ is the observed output variance.

A related problem involves finding the least achievable variance of the ith output, *i.e.*

$$J_{i,\text{decen}} = \min_{K_i} E[\text{tr}(y_i\, y_i^T)] \tag{4}$$

Similar to (3), the MV benchmark for the ith output can be defined as

$$\eta_{i,\text{decen}} = \frac{J_{i,\text{decen}}}{E[\text{tr}(y_i\, y_i^T)]} \tag{5}$$

where $E[\text{tr}(y_i\, y_i^T)]$ is the observed variance of the ith output.

3. LOOP-BY-LOOP ANALYSIS

We first consider performance assessment of the decentralized controller on a loop-by-loop basis. For clarity of presentation, we limit the discussion to 2×2 systems. The result can be generalized to $n_y \times n_y$ systems using block-partitioning of G and H. We have

$$y_1 = G_{11}\, u_1 + G_{12}\, u_2 + H_1\, a \tag{6}$$

$$y_2 = G_{21}\, u_1 + G_{22}\, u_2 + H_2\, a \tag{7}$$

The time delay associated with G_{ij} is denoted as d_{ij}, *i.e.*

$$G_{ij} = q^{-d_{ij}} \bar{G}_{ij} \tag{8}$$

where $\bar{G}_{ij}$ denotes the invertible part of G_{ij}. Without loss of generality, we consider that the objective is to characterize the least achievable variance of y_1.

3.1 Conventional approach

The traditional approach for loop-by-loop analysis involves using the MV benchmark for SISO systems. Here, H_1 is decomposed using Diophantine identity as

$$H_1 = F_1 + q^{-d_{11}} R_1 \tag{9}$$

Then, the least achievable variance of y_1 is taken as (Åström, 1970; Harris, 1989)

$$J_1 = \min_{K_1} E[\text{tr}(y_1\, y_1^T)] = \|F_1\|_2^2 \tag{10}$$

where $\|\cdot\|_2$ denotes the $\mathcal{H}_2$-norm. The MV benchmark for individual outputs is defined similar to (5). An inherent assumption in the derivation of (10) is that $u_2 = 0$ at all times or in other words, the first loop is being operated in isolation from the rest of the process. We next demonstrate that when the presence of other loops is accounted for, the first d_{11} elements of H_1 are not necessarily feedback invariant and thus the least achievable variance of y_1 can be lower than J_1 in (10).

3.2 Modified MV benchmark

Consider that the second loop is closed with $u_2 = -K_2 y_2$. Under partially closed loop conditions, we have (Skogestad and Postlethwaite, 2005)

$$y_1 = P_{11}\, u_1 + P_{d1}\, a \tag{11}$$

where

$$P_{11} = G_{11} - \frac{G_{12}K_2G_{21}}{1+G_{22}K_2},\; P_{d1} = H_1 - \frac{G_{12}K_2H_2}{1+G_{22}K_2} \tag{12}$$

Since $1/(1+G_{22}K_2)$ and K_2 are rational and invertible, it follows that the delay associated with P_{11} or the effective delay of the first loop is

$$d_1^{'} = \min(d_{11}, d_{12} + d_{21}) \tag{13}$$

Now, let P_{d1} be decomposed using Diophantine identity as

$$P_{d1} = F_1^{'} + q^{-d_1^{'}} R_1^{'} \tag{14}$$

Using (11) and (14), it follows that

$$y_1 = F_1^{'}\, a + q^{-d_1^{'}} (\bar{P}_{11}\, u_1 + R_1^{'}\, a) \tag{15}$$

where $P_{11} = q^{-d_1^{'}} \bar{P}_{11}$ and $\bar{P}_{11}$ denotes the invertible part of P_{11}. Since the first term in (15) cannot be affected by u_1 (invariant of K_1), it follows that

$$J_{1,\text{decen}} = \min_{K_1} E[\text{tr}(y_1\, y_1^T)] = \|F_1^{'}\|_2^2 \tag{16}$$

Note that $\|F_1^{'}\|_2^2$ represents the least achievable variance of y_1, when the presence of second loop is accounted for and can be used readily for performance assessment of decentralized controllers on a loop-by-loop basis.

Remark 1. In general, $J_{1,\text{decen}}$ may depend on K_2. This dependance, however, has no bearing on the loop-by-loop performance assessment, where the objective is to find the least achievable variance of y_i through tuning of K_i. If both controllers are allowed to be tuned simultaneously, using similar analysis as used in this section earlier, it can be shown that the first $d_1^{''} = \min(d_{11}, d_{12})$ impulse response coefficients of H_1 are feedback invariant and thus

$$\min_{K_1,K_2} E[\text{tr}(y_1\, y_1^T)] = \|F_1^{''}\|_2^2 \tag{17}$$

where $H_1 = F_1^{''} + q^{-d_1^{''}} R_1^{''}$. Note that the bound in (17) is independent of controller type (full multivariable or decentralized).

Though the result in (16) may seem entirely mathematical, a physical reasoning with this result can be associated by considering the block diagram of a 2×2 system shown in Figure 3. Here, u_1 can affect y_1 directly through G_{11}, but also through a parallel path involving G_{12} and G_{21} (shown with thick line in Figure 3). Thus, $d_1^{'} = \min(d_{11}, d_{12}+d_{21})$ represents the delay of the fastest path through which u_1 can affect y_1. In this sense, loop interaction can sometimes be beneficial for reducing output variance.

Example 2. To illustrate the findings of this section, we use the case study of a binary distillation column (Wood and Berry, 1973). The continuous-time model is discretized with a sampling time of 1 minute to get

$$G = \begin{bmatrix} \dfrac{0.744q^{-2}}{1-0.942q^{-1}} & \dfrac{-0.879q^{-4}}{1-0.954q^{-1}} \\ \dfrac{0.579q^{-8}}{1-0.912q^{-1}} & \dfrac{-1.302q^{-4}}{1-0.933q^{-1}} \end{bmatrix} \tag{18}$$

$$H = \begin{bmatrix} \dfrac{0.247q^{-9}}{1-0.935q^{-1}} \\ \dfrac{0.358q^{-4}}{1-0.927q^{-1}} \end{bmatrix} \tag{19}$$

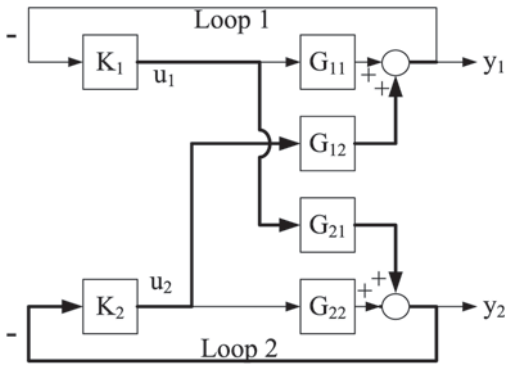


Fig. 3. Presence of parallel path (shown with thick line) from u_1 to y_1 for 2×2 systems

For diagonal pairings, the following decentralized PI controller is tuned using internal model control (IMC) method (Skogestad, 2003)

$$K = \text{diag}\left(\frac{0.652 - 0.571q^{-1}}{1 - q^{-1}}, \frac{-0.124 + 0.115q^{-1}}{1 - q^{-1}}\right) \quad (20)$$

which provides $\text{Var}(y_1) = 0.122$ and $\text{Var}(y_2) = 0.759$. After factoring the time delay of H, the least achievable variances of y_1 and y_2 according to the conventional approach discussed in Section 3.1 are $J_1 = 0.114$ and $J_2 = 0.413$, respectively. Thus, the conventional MV benchmarks for loops 1 and 2 are $\eta_1 = 0.932$ and $\eta_2 = 0.544$, respectively, which indicate that the variance of y_2 can be reduced significantly by re-tuning K_2, but tuning K_1 will not reduce variance of y_1 significantly. Using the modified MV benchmark, we next show that this conclusion is not entirely correct.

We have $d_{11} = 2$, $d_{12} = 4$, $d_{21} = 8$ and $d_{22} = 4$. Thus, $d_1^{'} = \min(d_{11}, d_{12} + d_{21}) = 2$ and $d_2^{'} = \min(d_{22}, d_{12} + d_{21}) = 4$. Since the first $d_2^{'} = d_{22}$ impulse response coefficients of P_{d2} and H_2 are the same, we find that $J_{2,\text{decen}} = J_2 = 0.413$ and $\eta_{2,\text{decen}} = \eta_2 = 0.544$. The first $d_1^{'} = d_{11}$ impulse response coefficients of P_{d1}, however, are different from the corresponding impulse response coefficients of H_1 and we find that $J_{1,\text{decen}} = 0.031$ and $\eta_{1,\text{decen}} = 0.251$. Thus, the modified MV benchmark identifies that the variance of y_1 can be reduced by approximately 4 times through tuning of K_1.

We point out that for this process, the first $d_1^{'}$ impulse response coefficients of P_{d1} and thus $J_{1,\text{decen}}$ depend on K_2. For example, when the gain of K_2 is decreased by a factor of 0.75, $J_{1,\text{decen}}$ increases to 0.037. With this controller tuning the variance of y_1 is 0.126. Thus, we have $\eta_{1,\text{decen}} = 0.293$, which indicates that the variance of y_1 can still be reduced by approximately 3 times.

Remark 3. Although for Example 2, the effective delays for both loops are the same as open-loop delays, *i.e.* $d_1^{'} = d_{11}$ and $d_2^{'} = d_{22}$, this is not true in general. For example, when pairings are chosen on the off-diagonal elements of G in (18), the effective delay for loop 1 is $d_1^{'} = \min(d_{12}, d_{11} + d_{22}) = 6$, which is different from open-loop delay, $d_{12} = 8$.

Remark 4. Based on (11), under closed loop conditions

$$y_1 = \frac{P_{d1}}{1 + P_{11}K_1} a \quad (21)$$

Let $K_1 = M_1/(1 - P_{11}M_1)$, where M_1 is a stable transfer function (Youla parameterization). Then,

$$y_1 = (1 - P_{11}M_1)P_{d1}\, a \quad (22)$$

$$= F_1^{'} a - q^{-d_1^{'}}(\bar{P}_{11}M_1P_{d1} - R_1^{'})a \quad (23)$$

Thus, $F_1^{'}$ and thus $\eta_{i,\text{decen}}$ can be estimated using closed loop data, *e.g.* using the FCOR algorithm (Huang and Shah, 1999), with *a priori* knowledge of the delays of G_{ij}. It is also interesting to note that the use of FCOR algorithm for loop-by-loop analysis of decentralized controllers leads to estimation of the first few impulse response coefficients of P_{d1} and not H_1, as is traditionally believed.

4. SIMULTANEOUS ANALYSIS

The variance of other outputs can increase, when the variance reduction of ith output is attempted through tuning of K_i. This effect is not taken into account by loop-by-loop analysis. To overcome this drawback, we derive an MV benchmark for simultaneous performance assessment of all loops in this section.

4.1 Conventional approach

A multivariable process can be represented as

$$G = D^{-1}\bar{G} \quad (24)$$

such that $\bar{G}$ and D^{-1} contain the invertible and non-invertible parts of G, respectively. We consider that $D(q)$ is a unitary interactor matrix, *i.e.* $D^T(q^{-1})\, D(q) = I$ (Huang and Shah, 1999). Let

$$q^{-d}DH = F + q^{-d}R \quad (25)$$

where d denotes the order of the interactor matrix. Then (Harris et al., 1996; Huang et al., 1997),

$$J_{\text{full}} = \min_K E[\text{tr}(y\, y^T)] = \|F\|_2^2 \quad (26)$$

The MV benchmark for simultaneous analysis is defined similar to (3). The bound on achievable output variance in (26), however, does not take the diagonal structure of the controller into account and thus may classify well-performing decentralized controllers as poorly performing. In the subsequent discussion, we present a lower bound on the achievable output variance for systems under decentralized control.

4.2 MV benchmark for Decentralized controllers

For regulatory control, we have $u = -K\, y$. Thus, the closed loop transfer function between a and y can be written as

$$y = S\, a; \quad S = (I + G\, K)^{-1}\, H \quad (27)$$

Since $E[a(t)a^T(t)] = I$,

$$J_{\text{decen}} = \min_K E[\text{tr}(y\, y^T)] = \min_K \|S\|_2^2 \quad (28)$$

When diagonal structure is imposed on K, the optimization problem in (28) becomes non-convex. A key observation to overcome this difficulty is that

$$\|S\|_2^2 = \sum_{i=0}^{\infty} \mathrm{tr}\,(S_i^T\, S_i) \geq \sum_{i=0}^{n} \mathrm{tr}\,(S_i^T\, S_i) \tag{29}$$

for any finite n. Thus, a lower bound on J_{decen} can be found by minimizing $\sum_{i=0}^{n} \mathrm{tr}\,(S_i^T\, S_i)$. Here, S_i represents the ith impulse response coefficient of S.

When the closed-loop system is stable, (27) can be expanded using Taylor Series expansion to get

$$S = \left[\sum_{i=0}^{\infty} (-1)^i (G\,K)^i\right] H \tag{30}$$

For given n, we define the following $nn_y \times nn_y$-dimensional Hankel matrices

$$G_H = \begin{bmatrix} G_0 & 0 & \cdots & 0 \\ G_1 & G_0 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ G_{n-1} & \cdots & G_1 & G_0 \end{bmatrix} \tag{31}$$

$$K_H = \begin{bmatrix} K_0 & 0 & \cdots & 0 \\ K_1 & K_0 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ K_{n-1} & \cdots & K_1 & K_0 \end{bmatrix} \tag{32}$$

and the following $nn_y \times nn_d$-dimensional matrices

$$H_v = \begin{bmatrix} H_0^T & H_1^T & \cdots & H_{n-1}^T \end{bmatrix}^T \tag{33}$$

$$S_v = \begin{bmatrix} S_0^T & S_1^T & \cdots & S_{n-1}^T \end{bmatrix}^T \tag{34}$$

Based on (32)-(34), S_v can be compactly written as

$$S_v = \left[\sum_{i=0}^{n} (-1)^i (G_H\, K_H)^i\right] H_v \tag{35}$$

Now, a lower bound on J_{decen} can be found by solving

$$J_{\mathrm{decen}} = \min_K \sum_{i=0}^{\infty} \mathrm{tr}\,(S_i^T\, S_i) \geq \min_K \mathrm{tr}\,(S_v^T\, S_v) \tag{36}$$

for any finite n. Based on (35), we note that S_v and thus $\mathrm{tr}\,(S_v^T\, S_v)$ depend polynomially on the controller parameters. Thus, the optimization problem in (36) can be seen as finding the global minimal value of a polynomial. For this purpose, we use sums of squares (SOS) programming (Parillo, 2000) in this paper. SOS programming transforms the polynomial minimization problem to a semi-definite program, which is solved using Sedumi (Sturm, 1999) interfaced with Matlab through Yalmip (Löfberg, 2004) in this paper.

SOS programming does not necessarily provide the minimum value of the polynomial, but guarantees a global lower bound (Parillo, 2000). For any controller K, however, since

$$\sum_{i=0}^{n+1} \mathrm{tr}\,(S_i^T\, S_i) \geq \sum_{i=0}^{n} \mathrm{tr}\,(S_i^T\, S_i) \tag{37}$$

tight lower bound on J_{decen} can be found by increasing n sequentially until convergence. In comparison with Kariwala (2007), where only the first $(2d-1)$ impulse response coefficients of S are used to find a lower bound on J_{decen}, the use of SOS programming provides a tighter lower bound, whenever $n > (2d-1)$. A similar approach involving SOS programming has been used earlier by Sendjaja and Kariwala (2009) to characterize the achievable output variance of SISO systems under PID control.

Example 5. We consider the following 2×2 process, where

$$G = \begin{bmatrix} \dfrac{-0.1q^{-2}}{(1-0.1q^{-1})\,(1-0.2q^{-1})} & \dfrac{-0.25q^{-1}\,(1-0.3q^{-1})}{(1-0.1q^{-1})\,(1-0.2q^{-1})} \\ \dfrac{0.5q^{-1}\,(1+0.9q^{-1})}{(1-0.1q^{-1})\,(1-0.2q^{-1})} & \dfrac{-0.1q^{-2}}{(1-0.1q^{-1})\,(1-0.2q^{-1})} \end{bmatrix} \tag{38}$$

and $H = 1/(1-q^{-1})\,I$ (Kariwala, 2007). Then, $D = q\,I$, $F = I$ and $J_{\mathrm{full}} = \|F\|_2^2 = 2$.

By considering the contribution of first $(2d-1)$ impulse response coefficients of S towards $\|S\|_2^2$, Kariwala (2007) found $J_{\mathrm{decen}} \geq 4$. Using the SOS programming approach, however, we find $J_{\mathrm{decen}} \geq 8.023$. Using non-convex optimization with multiple randomized initial guesses, Kariwala (2007) showed that the exact value of J_{decen} is approximately 8.16. This example amply demonstrates the usefulness of the SOS programming approach for finding tight lower bound on least achievable value of output variance for systems under decentralized control. The reader should also note that for this process, the MV benchmark found using conventional approach will be approximately 4 times lower than the MV benchmark found by accounting for the controller structure. Thus, the conventional approach for performance assessment of decentralized controllers may incorrectly classify well-performing controllers as poorly performing.

Remark 6. Unlike loop-by-loop analysis (see Remark 4), it is difficult to estimate η_{decen} directly from closed-loop data. When G is known (or has been identified using open or closed-loop identification experiments), H can be estimated by pre-whitening the pseudo-signal $(y - Gu)$. Then, SOS programming can be used to identify a lower bound on J_{decen} and η_{decen} based on identified model. We point out that the knowledge of G is also required by other available approaches for performance assessment of decentralized controllers (Ko and Edgar, 1998; Jain and Lakshminarayanan, 2007). In practice, the task of identifying G should be undertaken, only when J_{full} differs significantly from the observed output variance.

4.3 MV benchmark for Decentralized PID Controllers

In industrial practice, the individual sub-controllers of the decentralized controller are often fixed to be of PID-type. Clearly, the presence of additional controller structure can further limit the least achievable variance of outputs. Next, we show that the SOS programming approach can be easily extended to derive a tight lower bound on J_{decen} for decentralized PID controllers.

We note that the decentralized PID controller can be expressed as

$$K_{PID} = \frac{1}{\Delta} \sum_{i=0}^{2} C_i\, q^{-1} \tag{39}$$

where C_i has diagonal structure and $\Delta = 1 - q^{-1}$ is the integrator. For

$$\hat{G} = \frac{1}{\Delta} G \tag{40}$$

let us define the Hankel matrices $\hat{G}_H$ and C_H, which have the same structure as G_H and K_H defined in (32). Using similar approach, as used in Section 4.2, it can be shown that for any finite n

$$S_v = \left[\sum_{i=0}^{n} (-1)^i (\hat{G}_H\, C_H)^i\right] H_v \tag{41}$$

Thus, a lower bound on J_{decen} for decentralized PID controller can be found by minimizing $\text{tr}\,(S_v^T\, S_v)$ using SOS programming as before.

Example 7. We revisit Example 5. When individual sub-controllers are restricted to be of PI-type, the lower bound on J_{decen} increases to 9.980 (approximately 25% higher than unrestricted decentralized controller). The following sub-optimal controller is designed using trial and error

$$K_{\text{PI}} = \text{diag}\left(\frac{-0.629 + 0.474\, q^{-1}}{\Delta}, \frac{-2.844 + 1.862\, q^{-1}}{\Delta}\right) \tag{42}$$

which provides $E[\text{tr}(y\, y^T)] = 10.087$. When PID controllers are used, the lower bound on J_{decen} is 9.250 (approximately 15% higher than unrestricted decentralized controller). The following sub-optimal controller

$$K_{\text{PID}} = \text{diag}\left(\frac{-0.884 + 0.924\, q^{-1} - 0.205\, q^{-2}}{\Delta}, \frac{-3.210 + 2.764\, q^{-1} - 0.869\, q^{-2}}{\Delta}\right) \tag{43}$$

provides $E[\text{tr}(y\, y^T)] = 9.421$. For both cases (PI and PID control), the lower bounds are close to the upper bounds, which demonstrates that SOS programming technique can be used to find tight bounds on J_{decen} for decentralized PID controllers efficiently and reliably.

5. CONCLUSIONS AND OPEN ISSUES

The use of decentralized or multi-loop controllers is common in process industries. In this paper, we have shown that the use of existing MV benchmarks for SISO and MIMO systems for performance assessment of decentralized controllers may lead to incorrect conclusions regarding the opportunities for variance reduction through controller retuning. We proposed modified MV benchmarks for loop-by-loop analysis of the decentralized controller, which can be directly estimated from closed-loop data. An MV benchmark for simultaneous analysis of the decentralized controller is also proposed through the novel use of SOS programming, which guarantees a lower bound on the least achievable output variance. In summary, this paper takes a major step towards systematic performance assessment of decentralized controllers.

A limitation of the use of SOS programming approach is that the knowledge of process model is required. Furthermore, SOS programming requires solving large semi-definite programs, whose size and solution time increase rapidly with process dimensions. We are currently exploring the use of alternate approaches to handle the computational complexity of the SOS programming approach.

REFERENCES

Åström, K.J. (1970). *Introduction to Stochastic Control Theory*. Academic Press, London.

Harris, T.J. (1989). Assessment of control loop performance. *Can. J. Chem. Eng.*, 67(5), 856–861.

Harris, T.J., Boudreau, F., and Macgregor, J.F. (1996). Performance assessment of multivariable feedback controllers. *Automatica*, 32(11), 1505–1518.

Huang, B. and Shah, S.L. (1999). *Performance Assessment of Control Loops: Theory and Applications*. Springer-Verlag, London.

Huang, B., Shah, S.L., and Kwok, E.K. (1997). Good, bad or optimal? Performance assessment of multivariate processes. *Automatica*, 33(6), 1175–1183.

Jain, M. and Lakshminarayanan, S. (2007). Estimating performance enhancement with alternate control strategies for multiloop control system. *Chem. Eng. Sci.*, 62, 4644–4658.

Jelali, M. (2006). An overview of control performance assessment technology and industrial applications. *Control Eng. Pract.*, 14(5), 441–466.

Kariwala, V. (2007). Fundamental limitation on achievable decentralized performance. *Automatica*, 43(10), 1849–1854.

Kariwala, V., Forbes, J.F., and Meadows, E.S. (2005). Minimum variance benchmark for decentralized controllers. In *Proc. American Control Conference*, 1437–1442. Portland, OR, USA.

Ko, B. and Edgar, T.F. (1998). Multiloop PID control performance monitoring. In *Proc. The 8th Congress of Aston Pacific Confederation of Chemical Engineering*. Seoul, Korea.

Löfberg, J. (2004). YALMIP : A toolbox for modeling and optimization in MATLAB. In *Proc. the CACSD Conference*. Taipei, Taiwan.

Parillo, P.A. (2000). *Structured Semidefinite Programs and Semialgebraic Geometry Methods in Robustness and Optimization*. Ph.D. thesis, California Institute of Technology, Pasadena, California, USA.

Qin, S.J. (1998). Controller performance monitoring - A review and assessment. *Comput. Chem. Eng.*, 23, 173–186.

Rotkowitz, M. and Lall, S. (2006). A characterization of convex problems in decentralized control. *IEEE T. Automat. Contr.*, 51(2), 274–286.

Sendjaja, A.Y. and Kariwala, V. (2009). Achievable PID performance using sums of squares programming. *J. Proc. Contr.*, In press.

Skogestad, S. (2003). Simple analytic rules for model reduction and PID controller tuning. *J. Proc. Contr.*, 13(4), 291–309.

Skogestad, S. and Postlethwaite, I. (2005). *Multivariable Feedback Control: Analysis and Design*. John Wiley & Sons, Chichester, UK, 2nd edition.

Sourlas, D.D. and Manousiouthakis, V. (1995). Best achievable decentralized performance. *IEEE T. Automat. Contr.*, 40(11), 1858–1871.

Sturm, J. (1999). Using SeDuMi 1.02, A Matlab toolbox for optimization over symmetric cones. *Optim. Method Softw.*, 11(1), 625–653.

Wood, R.K. and Berry, M.W. (1973). Terminal composition control of a binary distillation column. *Chem. Eng. Sci.*, 28(9), 1707–1717.

Yuz, J.I. and Goodwin, G.C. (2003). Loop performance assessment for decentralized control of stable linear systems. *Eur. J. Control*, 9(1), 116–130.

Eliminating Valve Stiction Nonlinearities for Control Performance Assessment ⋆

W. Yu * **D.I. Wilson** ** **B.R. Young** ***

* *Industrial information & Control Centre, The University of Auckland, Auckland, New Zealand (e-mail:* w.yu@auckland.ac.nz*).*
** *Electrical & Electronic Engineering, Auckland University of Technology, Auckland, New Zealand (e-mail:* david.i.wilson@aut.ac.nz*)*
*** *Depart. of Chemical & Materials Engineering, The University of Auckland, Auckland, New Zealand, (e-mail:* b.young@auckland.ac.nz*)*

Abstract: Control performance assessment or CPA is a useful tool to establish the quality of industrial feedback control loops, but this requires establishing the minimum variance lower bound. While reliable algorithms have been developed for linear systems, common nonlinearities such as valve stiction require modifications to the basic strategy.
If the valve gets stuck due to stiction, for stable plants the output will reach steady state until the valve again moves. During this time the nonlinearity due to stiction is essentially removed from the system, and it is possible to compute performance assessment indices in the standard manner.
This paper describes an automated strategy to reliably identify these linear steady-state periods and subsequently compute the minimum variance lower bounds. The results of a simulation example illustrate that the proposed methodology is efficient and accurate enough for the classes of systems and nonlinearities considered to provide statistics for control performance assessment for linear systems with nonlinearities caused by valves.

Keywords: Control performance assessment, Valve stiction, Steady state.

1. INTRODUCTION

It is perhaps not surprising that instrument and control engineers are overwhelmed by the sheer number of loops that need attention on any typical industrial processing plant. Many loops are mis-tuned, if tuned at all, as noted by Bialkowski (1998) and other practitioners, and many control valves are only maintained when something catastrophic occurs. However the economic benefits from improving the performance of control loops, even those operating at a cursory glance acceptably, is often grossly under estimated.

In this paper, we present a strategy to compute the minimum variance lower bound, which is arguably the difficult step in quantifying the performance improvement of a typical control loop that suffers from the specific nonlinear phenomena of valve stiction being a very common cause for poor control performance.

Control performance assessment, or CPA, is a technology to diagnose and maintain operational efficiency of control systems. CPA is routinely applied in the refining, petrochemicals, pulp and paper and the mineral processing industry as noted by Qin (1998), Harris (1999), Huang and Shah (1999), Jelali (2006), although these, and many other publications, are mainly restricted to linear systems.

In the case of nonlinear systems, Harris and Yu (2007) superimposes the nonlinear dynamic model to an additive linear or partially nonlinear disturbance. It is shown that a minimum variance feedback invariant exists and the minimum variance performance can be estimated from routine operating data. Continuing this idea, a semi-parametric method was proposed in Yu et al. (2008) to find the minimum variance lower bound for linear systems with valve stiction. In that work a local smoothing spline approximated the stiction nonlinearity, but given the complexity of the nonlinearity, and the heuristic approach, it must be expected that this strategy will fail for some cases.

In this paper, we will extend CPA to a important practical nonlinear problem, that of control valve stiction. The performance degradation due to stiction prompted Horch (1999), Choudhury et al. (2005, 2006), Thornhill and Horch (2007) to investigate ways to diagnose the issue, while Jelali (2008) and the references therein, attempt the estimation of parametric stiction models, but few have continued the analysis to quantify the performance loss. Consequently, rather than attempt to approximate the nonlinearity, the approach taken here is to develop an automated strategy that extracts the steady state periods resultant once the valve is stuck fast. Based

⋆ Sponsor and financial support from the Industrial Information and Control Centre, AUT University and The University of Auckland, New Zealand.

on readily available input/output data collected during these periods, the minimum variance lower performance bound is computed in the standard manner. This gives an indication of how the control loop, even one suffering from stiction, would perform if it was serviced. Of course this presupposes that one is not allowed the luxury of setting the valve under consideration in manual, or one is charged with assessing many hundreds of operating loops.

The incentive to compute this control performance index is that it delivers a benchmark giving the engineer an idea of the improvement potential if the valve was to be serviced. For example, it could well be that of the hundreds of valves on site that required maintenance, the expected performance improvement, even if the stiction was entirely removed, would not be worth the time and effort.

The layout of the paper is as follows. In Section 2, the problem statement and model including valve stiction is introduced. Section 3 describes the methodology proposed in this paper to first extract, then check the validity of the steady-state linear periods. Section 4 illustrates by way of simulation the performance of the proposed methodology. This is followed by a discussion and conclusions highlighting both the limitations and potential of the method.

2. PROCESS DESCRIPTION

We assume the plant can be adequately modelled by

$$y_t = \frac{B(q^{-1})}{A(q^{-1})} q^{-b} u_t + d_t \tag{1}$$

where $A(q^{-1})$ and $B(q^{-1})$ are polynomials in the backshift operator q^{-1}, and b is the time delay of the system. The disturbance d_t is modelled as the output of a linear Autoregressive-Integrated-Moving-Average (ARIMA) filter driven by white noise a_t of zero mean and variance σ_a^2 of the form

$$d_t = \frac{\theta(q^{-1})}{\phi(q^{-1})\nabla^d} a_t = \psi(q^{-1}) a_t \tag{2}$$

where $\nabla \stackrel{\text{def}}{=} (1 - q^{-1})$ is the difference operator and d is a non-negative integer, typically less than 2. The polynomials $\theta(q^{-1})$ and $\phi(q^{-1})$ are monic and stable.

A common process nonlinearity afflicting control valves is known as 'stiction' which exhibits a range of of nonlinear behaviour including hysteresis, backslash and deadzones, both dynamic and static, and is summarised in de Wit et al. (1995), Lampaert et al. (2004).

Fig. 1 illustrates the typical sawtooth characteristic behaviour of a poorly maintained valve suffering from stick/slip friction using the stiction model developed in Choudhury et al. (2004). It is important to note that under normal industrial operations, the manipulated variable (MV) signal injected into the plant, here denoted as u^v, is unobservable.

We include the nonlinear stiction function $f(\cdot)$ (which represents the relationship between the manipulated variable and the actual valve output), into Eqn. 1 giving

$$y_t = \frac{B(q^{-1})}{A(q^{-1})} q^{-b} f(u_t) + d_t \tag{3}$$

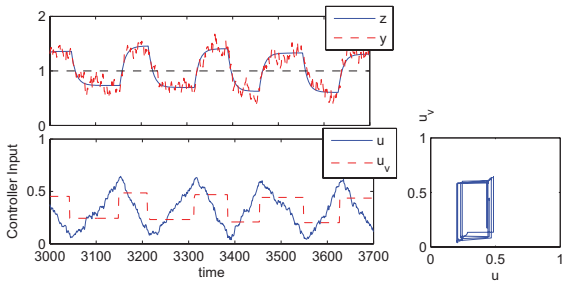


Fig. 1. The output of the controller, u, and the subsequent output of the control valve, u^v, suffering from slip/stick stiction.

Fig. 2 summarises the system considered in this paper. The intended controller output, u, (sometimes referred to as OP), is typically different from the actual valve position, u^v, due to the stiction. In the ideal case however, we can simply assume $u_t^v = f(u_t) = u_t$.

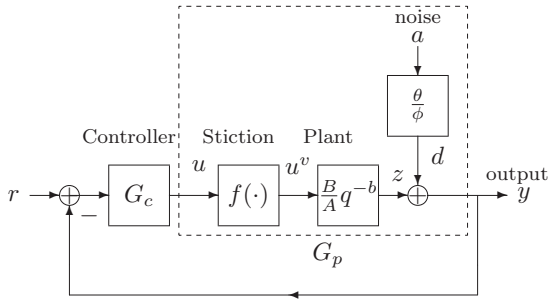


Fig. 2. Closed loop system with valve stiction under consideration

3. MINIMUM VARIANCE PERFORMANCE BOUNDS: VALVE STICTION CASES

The basis for minimum variance performance bounds was developed by Harris (1989) where it was was shown that the minimum variance performance bound for linear systems could be estimated from routine closed-loop data provided the process delay is known in advance. For the system described by Eqn. 1, the minimum variance performance lower bound is simply

$$\sigma_{MV}^2 = (1 + \varphi_1^2 + \cdots + \varphi_{b-1}^2)\sigma_a^2 \tag{4}$$

where the φ weights are the first $b-1$ impulse coefficients of the disturbance transfer function in Eqn. 2.

The minimum variance performance bounds for a class of nonlinear systems described by Eqn. 3 have been reported in Harris and Yu (2007) which used a nonlinear polynomial-AR or polynomial-ARX model to estimate the b-step ahead prediction. The drawback for this application is that it is difficult to find a general function to adequately approximate the valve stiction. Notwithstanding, the non-parametric spline method to approximate the nonlinearity proposed in Yu et al. (2008) partially overcomes the issue of modelling valve stiction/friction, but it too will fail for some cases.

In this paper, we propose a method to find the minimum variance performance bounds for valve stiction cases. Rather than trying to find a parametric or non-parametric function to approximate the nonlinear function as suggested in Yu et al. (2008), we will focus solely on the periods when we know, or at least suspect, that the system is operating in a linear regime. That way, we can simply ignore the now non-existent nonlinearity, and compute the minimum variance lower bound in the standard manner. The success of this strategy depends on how well we can establish that the system is behaving essentially as a linear system.

We can potentially ignore the effects of the nonlinearity by exploiting a unique characteristic of valve stiction. Due to the stick/slip friction, the times that the valve is stuck gives the system a chance to reach steady state and during these periods, we can use linear ARMA techniques to estimate the lower performance bound.

The key problem is how to identify the steady state periods from the closed loop output data. Our approach includes two parts. First we use a heuristic pattern method to select the periods of steady state in the observable time series y, and we validate this by employing a linearity test. Second, given possibly multiple segments of a linear time series, not necessarily contiguous, we can now fit a linear ARMA model and subsequently compute the minimum variance performance bound. The details of the methodology are discussed in the following sections.

3.1 Identifying steady-state periods

The presence of valve stiction induces a limit cycle with a characteristic triangular shape in the controller output as is shown in Fig. 1. This cycling is exacerbated by the integral component of the controller which eventually increases to such an extent that the stuck valve again moves. Unfortunately the valve moves too far, and the cycle begins again as demonstrated in Fig. 3.

Since the information of the actual valve output position, u^v, is not available for most industrial implementations, (if it was, we could simply use this series for the computation), the steady state periods must be identified from the data consisting solely of the process output y and controller output u.

Under the cycling conditions due to stiction, the scaled difference between u and y will describe a sawtooth trend as given in Fig. 4. The discontinuous turning points of the triangle wave indicate when the valve actually moves and it is these instances, (depicted as the vertical dashed and dotted lines) that we need to identify.

Given that most industrial plants have a stable low-pass frequency response that attenuates high frequency noise, and that the plant measurement is disturbed by d, it may be necessary to weight the subtraction to better highlight the trend. Fig. 4 actually trends $cu - y$ where the scale factor c is in this case 10.

After the periods of steady state are identified, an initial segment in each period will be discarded so that the previous input effects will be removed. Then subsequently

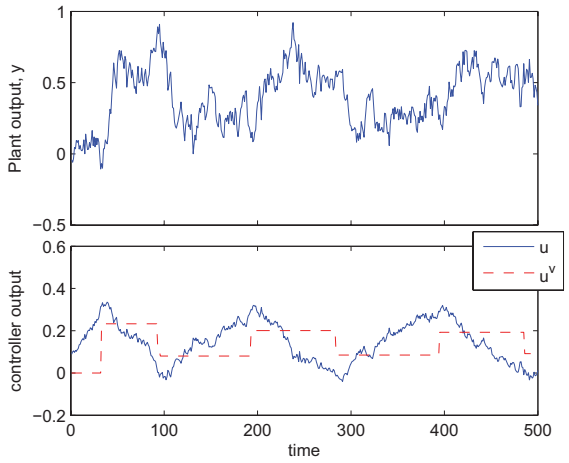


Fig. 3. The plant output, y, the controller output u, and the output of the valve suffering from stiction, u^v.

we can derive a minimum variance controller performance lower bound using these periods of data.

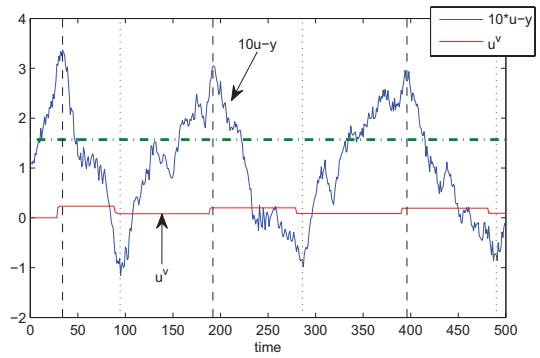


Fig. 4. Establishing the steady-state periods by computing $cu - y$.

An automated procedure to establish the stuck periods relies on reliably identifying the maxima and minima of the sawtooth shaped $cu - y$ trend in Fig. 4. First the times of the zero-crossings and their directions (falling, or raising edges) are established. Then, between each crossing instance, a search is made for the corresponding maximum or minimum.

Of course using such a heuristic approach, this simple algorithm is suspectable to false positives and correspondingly derives estimates of the steady-state periods shorter than the actual period. In cases of excessive suspected false crossings, standard techniques such as data smoothing or a Fourier identification of the dominant period could be applied to the noisy data series. However since these erroneous short periods will not be used for the minimum variance calculation anyway, they do not overly deteriorate the quality of the computed result. They do however, lower the efficiency of the data use.

3.2 Ensuring the removal of any nonlinearities

Notwithstanding the expectation that the valve stiction nonlinearity exhibits little memory, we need to be assured

that the selected period is in fact linear. One way is to do this is to apply a statistical test of linearity proposed by Subba and Gabar (1980), Hinich (1982) and previously used in this context by Choudhury et al. (2004) and Yu et al. (2008). This test, known as the Hinich test, is both nonparametric and reasonably robust.

In the simulations subsequently presented in section 4 it was obvious that the periods were linear so the nonlinear test was not actually employed. However if one is still in doubt, Yu et al. (2008) illustrates how such a validation could be performed.

3.3 Establishing the limit cycle

The proposed strategy works best when the valve is stuck for relatively long periods allowing the system to reach steady state. That is, periods larger than about 10 dominant time constants since we discard the first 3–4 to allow the system to reach steady-state, and then use the remaining data for the ARMAX model identification. Consequently we desire that the period of oscillation due to the valve nonlinearity is long compared to the settling time of the compensated loop and therefore we need to *a priori* establish reasonable conditions when that is likely to occur.

From the literature, and our simulation experience detailed further in section 4, it is found that there are three main factors which will affect the period of oscillation, namely the tuning of the PI controller, the magnitude of the disturbance, and the valve characteristics of the valve stiction.

4. SIMULATION EXPERIMENTS

The purpose of this section is to demonstrate the proposed method for the minimum variance performance assessment for valve stiction cases. A second order single-input, single-output (SISO) system with time constants 10 and 2, and steady-state gain of 3 is sampled at $T_s = 1$ to give

$$G_p = \frac{B}{A} = \frac{0.04338 + 0.03755q^{-1}}{1 - 1.621q^{-1} + 0.6483q^{-2}} \tag{5}$$

with time delay $b = 4$ under feedback control with controller

$$G_c = \frac{0.11 - 0.1q^{-1}}{1 - q^{-1}} \tag{6}$$

was used for generating simulated data. An additive disturbance of

$$d_t = \frac{0.2a_t}{1 - 0.8q^{-1}} \tag{7}$$

where a_t is a sequence of independent and identically distributed Gaussian random variables with zero mean and nominal variance $\sigma_a^2 = 0.1$.

A data-driven model for valve stiction proposed by Choudhury et al. (2004) is used to simulate the valve stiction. The model is characterised by two parameters, s for the valve stickiness and j for the magnitude of the valve jump. The closed loop behaviour with various combinations of s and j are plotted in Fig. 5. Pure deadband occurs when $j = 0$, (b) represents the undershoot case of a sticky valve, $s < j$, (c) illustrates the pure stick-slip situation, $s = j$ and (d) shows the valve output overshooting case, $s < j$. Note that the oscillation period decreases while the j value increases.

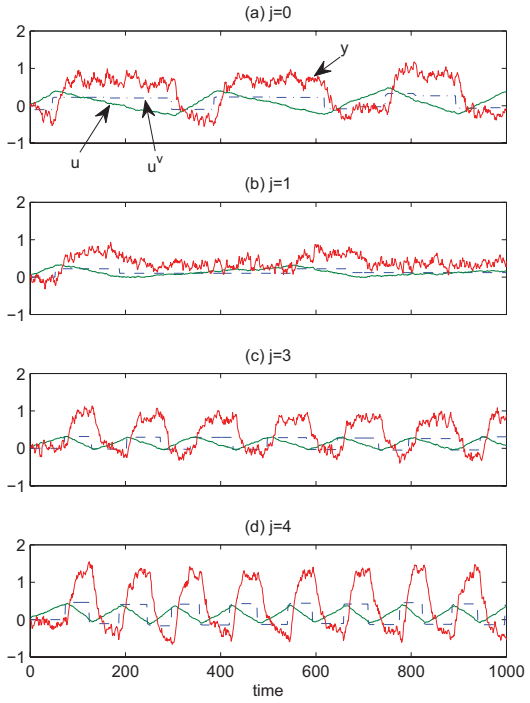


Fig. 5. The closed-loop behavior for $s = 3$ and with various values of j: (a) $j = 0$ (b) $j = 1$ (c) $j = 3$ (d) $j = 4$.

As noted in the pervious section, this strategy is reliant on relatively long periods when the valve is stuck. Given a fixed PI controller, we can vary the magnitude of the disturbance and the stiction jump/slip parameters, j, s and use a Monte-Carlo simulation to establish the largest period on average of the oscillation for each (σ_a^2, s, j) triplet. The resultant contour plots of periods are shown in Fig. 6.

In all cases we are most interested in the 'islands' of high periods apparent in all three examples given in Fig. 6.

Areas with periods less than about 100 are not interesting and are not plotted in Fig. 6. This is because since the plant in Fig. 6 has a dominant time constant of 10 sampled at 1, ten dominant time constants correspond to about 100 samples. We discard 30 to 40 samples to allow time for the system to reach steady-state, leaving a minimum of 60 samples in which to do the ARMA model identification. This data series length is about at the minimum recommended by Ljung (1987).

As expected the 'islands' of large periods occur when both the noise, σ_a^2, and jump parameters are not too big. This makes sense because if both s and j are zero, there is no stiction, and there is no self sustained oscillation due to the nonlinearity. Also for a given stiction, as the noise variance increases beyond the deadband limit, the valve will continually move, lowering, or completely eliminating, the potential steady-state periods.

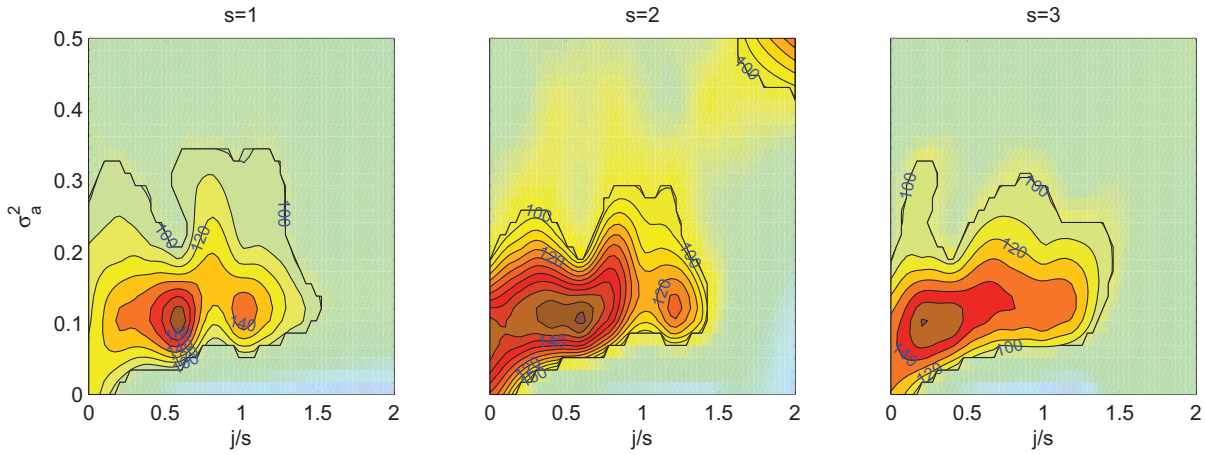


Fig. 6. The period of oscillation as a function stiction jump parameter j, and noise disturbance variance σ_a^2, at differing levels of stiction slip parameter s. Areas where the period is less than about 100 samples are not interesting for t his specific application because they are too short to perform a meaningful identification.

Due to the intractable nature of the nonlinearity, a Monte Carlo method is used to estimate the performance of the proposed strategy to estimate the minimum variance, σ_{MV}^2. 1000 observations generated from the valve stiction simulation are passed to the automated steady-state period identifier described in section 3.1 from which suitable periods are extracted. An ARMA model is fitted to the longest period from which σ_{MV}^2 is directly computed from the parametric model. This procedure is repeated 500 times.

The estimates of σ_{MV}^2 and associated uncertainties for different valve slip/jump conditions are shown in the comparative box plot in Fig. 7 again as in Fig. 5 for the case where $s = 3$ and various values for j. The true value of the minimum performance lower bound for this example is $\sigma_{MV}^2 = 9.2 \times 10^{-3}$.

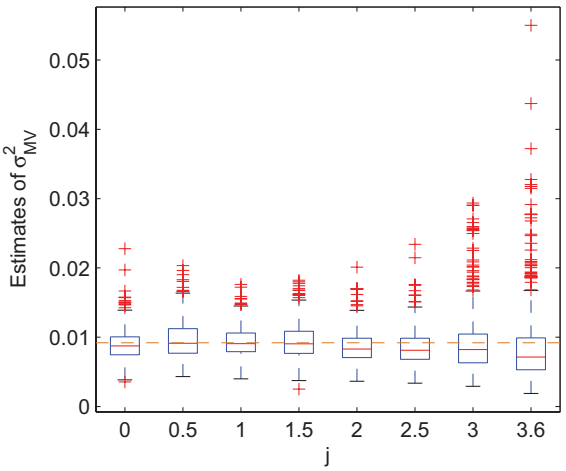


Fig. 7. Comparative box plots of the quality estimates of σ_{MV}^2 for $s = 3$ and for different j models. The horizontal dashed line is the true value of $\sigma_{MV}^2 = 9.2 \cdot 10^{-3}$.

5. DISCUSSION

The results from the numerical experiments to establish the minimum variance performance lower bound from normal operating data given in Fig. 7 show that the proposed strategy does reconstruct the correct σ_{MV}^2. Furthermore it is interesting to note that the quality of the estimate is best for the jump parameter $j \approx 1$, while for values smaller, and particularly larger, it begins to deteriorate. This is consistent with the results presented in Fig. 6 reinforcing the requirement to have reasonably long periods of steady-state to extract statistically significant results.

In the cases where the jump parameter j is larger than the slip s, we experience short periods of the stuck valve coupled with a comparatively large nonlinearity that contributes to the obvious deterioration in the confidence of the estimated minimum variance lower bound.

Similarly, if the zero-crossings are too frequent (and therefore the period available for steady-state consideration is too short compared to the anticipated dominant time constant), then we suggest that the strategy proposed in Yu et al. (2008) which uses smoothing splines to remove the nonlinearity might be more appropriate.

The proposed strategy has some limitations. First of all, as developed, we assume that the plant is stable, and reasonably well-damped. For type 1 plants with integrators, it is of course possible to differentiate the output. Furthermore, we assume that the dominant time constant is approximately known in order to discard the appropriate amount of data while waiting for steady-state. This is unlikely to be an overly onerous requirement for any processing plant. Finally we restrict our attention to those cases with moderate extent of stiction, since of course excessive stiction *must* be addressed, and minimal stiction would probably not be noticed anyway.

6. CONCLUSIONS

Valve stiction is a debilitating feature of many control loops that cannot, nor should not, be corrected by con-

troller tuning. However given the time and energy required to service the valve, it may be prudent for the instrument engineer to first establish what the best controlled performance would be if the valve was serviced.

The strategy proposed in this paper establishes the minimum performance lower bound in the case of excessive valve stiction using only observable signals and estimates of the plant dominant time constants and plant delay. In the case of rapid oscillation in the limit cycle, it is possible to stitch the short periods together to build up enough input/output data to make a reasonable identification.

While the examples considered only nonlinearities introduced by valve stiction, this strategy will work for any system which reaches steady states and stays there for a while.

ACKNOWLEDGMENTS:

Financial support to this project from the Industrial Information and Control Centre, Faculty of Engineering, The University of Auckland, New Zealand is gratefully acknowledged.

REFERENCES

W. L. Bialkowski. Dreams versus reality: A view from both sides of the gap. *Pulp & Paper Canada*, 94(11):19, 1998.

M.A.A.S. Choudhury, N.F. Thornhill, and S.L. Shah. Diagnosis of poor control loop performance using higher order statistics. *Automatica*, 40(10):1719–1728, 2004.

M.A.A.S. Choudhury, N.F. Thornhill, and S.L. Shah. Modelling valve stiction. *Chemical Engineering Progress*, 13(5):641–658, 2005.

M.A.A.S. Choudhury, S.L. Shah, N.F. Thornhill, and D.S. Shook. Automatic detection and quantification of stiction in control valves. *Control Engineering Practice*, 14:1395–1412, 2006.

C. Canudas de Wit, H. Olsson, K. J. Åström, and P. Lischinsky. A New Model for Control of Systems with Friction. *IEEE transactions on automatic control*, 40(3):419–425, March 1995.

T.J. Harris. Assessment of control loop performance. *Canadian Journal of Chemical Engineering*, 67:856–861, 1989.

T.J. Harris. A review of performance monitoring and assessment techniques for univariate and multivariate control systems. *J. Process Control*, 9(1):1–17, 1999.

T.J. Harris and W. Yu. Controller assessment for a class of nonlinear systems. *J. Process Control*, 17:607–619, 2007.

Melvin J. Hinich. Testing for Gaussianity and linearity of a stationary time series. *Journal of Time Series Analysis*, 3(13):169–176, 1982.

A. Horch. A simple method for detection of stiction in control valves. *Control Engineering Practice*, 7:1221–1231, 1999.

B. Huang and S.L. Shah. *Performance Assessment of Control Loops: Theory and Applications*. Springer, 1999.

Mohieddine Jelali. An overview of control performance assessment technology and industrial applications. *Control Engineering Practice*, 14(5):441–466, 2006.

Mohieddine Jelali. Estimation of valve stiction in control loops using separable least-squares and global search algorithms. *Journal of Process Control*, 18(7–8):632–642, 2008. ISSN 0959-1524.

Vincent Lampaert, Jan Swevers, and Farid Al-Bender. Comparison of model and non-model based friction compensation techniques in the neighbourhood of pre-sliding friction. In *Proceedings of the 2004 American Control Conference*, pages 1121–1126, Boston, Massachusetts, June 30 – July 2 2004.

Lennart Ljung. *System Identification: Theory for the User*. Prentice–Hall, 1987.

S. Joe Qin. Control performance monitoring – A review and assessment. *Computers in Chemical Engineering*, 23(2):173–186, 1998.

R.T. Subba and M.M. Gabar. A test for linearity of stationary time series. *Journal of Time Series Analysis*, 1(2):145–158, 1980.

Nina F. Thornhill and Alexander Horch. Advances and new directions in plant-wide disturbance detection and diagnosis. *Control Engineering Practice*, 15:1196–1206, 2007.

Wei Yu, David I. Wilson, and Brent R. Young. Control performance assessment in the presence of valve stiction. In K. Grigoriadis, editor, *The Eleventh IASTED International Conference on Intelligent Systems and Control, ISC 2008*, pages 379–384, Orlando, Florida, USA, 16–18 November 2008. ISBN 978-0-88986-777-2.

Valve Stiction Evaluation Using Global Optimization

M. Farenzena and J. O. Trierweiler

Group of Intensification, Modelling, Simulation, Control and Optimization of Processes (GIMSCOP)
Department of Chemical Engineering, Federal University of Rio Grande do Sul (UFRGS)
Rua Luiz Englert, s/n CEP: 90.040-040 - Porto Alegre - RS - BRAZIL,
Fax: +55 51 3308 3277, Phone: +55 51 3308 4163
E-MAIL: {farenz, jorge}@enq.ufrgs.br

Abstract: Valve stiction is a well known villain in process industry. Quantifying this valve damage is essential to ensure plant stability and profitability. The scope of this work is to propose a new method to compute valve stiction parameters, using a two parameter model, using only routine operating data. The proposed method uses global optimization to evaluate loop and plant parameters. Combining the proposed procedure with an efficient global optimization algorithm, the mean computation time for each valve was about 5 minutes. The method was applied in both simulation and industrial valves, providing reliable results, with relative errors smaller than 3% in all parameters.

Keywords: Performance Monitoring, Valve, Static Friction, Hysteresis, Global Optimization.

1 INTRODUCTION

In the last two decades, control loop performance monitoring has been a fruitful research field, providing automatic tools for process industry, which has great interest in evaluating loop performance in real time. Inside this scope, one topic that has been in focus is valve stiction, whose frequency in control loops is about 30% (Bialkowski, 1993). The effects of stiction are one explanation for these developments: it can cause plant-wide oscillations and increase the variability of the process and products.

Evaluating loop stiction is not a new issue. First studies were dated from 60's (Brown, 1958). However, in the last years, a big effort has been made to diagnose and solve this valve illness. A first group of works aimed at diagnosing stiction automatically, using only process variable (PV) and controller output (OP) (He *et al.*, 2007, Horch, 1999, Rossi and Scali, 2005, Ruel, 2000, Scali and Ghelardoni, 2008, Singhal and Salsbury, 2005, Stenman *et al.*, 2003, Yamashita, 2006). Some works have proposed specific stiction models for diaphragm type valves (Chen *et al.*, 2008, Choudhury *et al.*, 2005). A good survey about stiction models was recently written by Garcia (2008).

Also, some authors have proposed to compute stiction parameters in real time. Choudhury et al. (2006) proposed two methods to quantify stiction parameters, based on ellipse fitting and *c-clustering*, using a 1 parameter empirical model. Subsequently, some authors have proposed to quantify stiction parameters using a more reliable model, with two parameters. Choudhury *et al.* (2008) have proposed a method based on optimization and grid search.

Recently, Jelali (2008) has introduced one methodology to compute both stiction models, based on least-squares and global search algorithms. This method has two drawbacks: it is dependent of initial guess and it is computationally expensive. The scope of this work is to propose an alternative method based on global optimization to compute stiction parameters and linear plant model.

The main difference between our and Jelali's method is the optimization procedure, which is made in a single step, using global optimization. Both plant models and stiction parameters are computed in each optimization step. Combining the proposed procedure with an efficient global optimization algorithm, the computation time for each valve was less than 5 minutes.

The proposed methodology was applied in a set of 1000 simulation valves, with a relative error smaller than 3% in all cases, for all parameters. Then, the proposed method was applied in a group of industrial valves, showing reliable results.

The paper has been organized as follows: in section 2, the stiction definition, model and methodologies to evaluate valve stiction will be briefly discussed. Then, in section 3, the proposed methodology will be detailed. Several simulated and industrial case studies are shown in section 4, to corroborate the applicability of the proposed methodology. The paper ends with the concluding remarks.

2 STICTION: MODEL AND COMPUTATION

Stiction, or high static friction, can be defined as the valve damage that keeps the stem from moving, because the static friction exceeds the dynamic. As a consequence, the force to move the steam is generally larger than the desired new stem value, and the movement is jumpy (Ruel, 2000).

2.1 Stiction: model

A valve "suffering from stiction" has in the phase plot MV versus OP, shown in Fig. 1, four components: deadband (*DB*), stickband (*SB*), slip jump (*J*) and moving phase (*MP*). The method assumes that the process and controller have

linear behaviour, while the sticky valve inserts in the loop nonlinear behaviour.

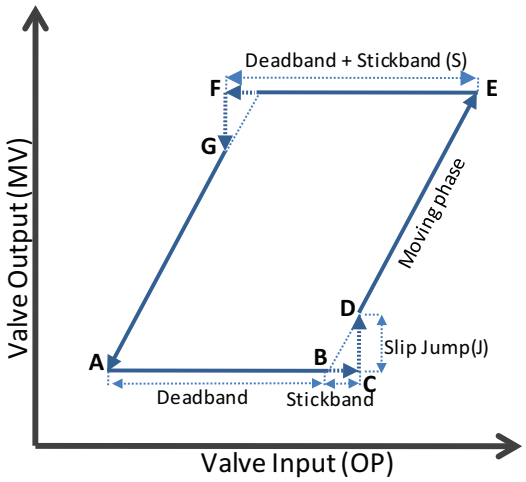


Fig. 1. Relation between controller output (OP) and valve position (MV) for a sticky valve.

When the valve changes the direction (A), the valve becomes sticky. The controller should overcome the deadband (AB) plus stickband (BC), and then the valve jumps to a new position (D). The stiction model consists of these two parameters: *S* (deadband+stickband) and *J* (jump).

Next, the valve starts moving, until its direction changes again or the valve comes to rest, between D and E.

The deadband and stickband represent the behaviour of the valve when it is not moving, although the input of the valve keeps changing. Slip jump represents the abrupt release of potential energy stored in the actuator chambers due to high static friction in the form of kinetic energy as the valve starts to move. The magnitude of the slip jump is crucial to determine the limit cycle amplitude and frequency.

The stiction model used in this work is proposed by Kano (2004), which is an extension of Choudhury's method, where stiction is modeled using two parameters. Their main advantage is that it can deal with both stochastic and deterministic signals. Kano's model flowchart representation is shown in Fig. 2.

The first two branches check the valve bounds. In the Kano's model, two valve states are distinguished: moving (*stp=0*) or resting (*stp=1*). When the valve changes its direction, its actual position state (us) is kept, until the static force is overcome. The friction force direction is denoted by $\pm d$.

2.2 Stiction: computation

In the literature, two methods to compute stiction parameters, using only normal operating data are proposed.

In the method proposed by Jelali (2008), a two step procedure is described. In the first step the stiction parameters are quantified using pattern search methods or genetic algorithms (GA). Next, the low-order linear plant model is identified, using a least-squares estimator. Both simulation and industrial valves are analyzed, and the errors between predicted and actual values for stiction parameters are less than 10%.

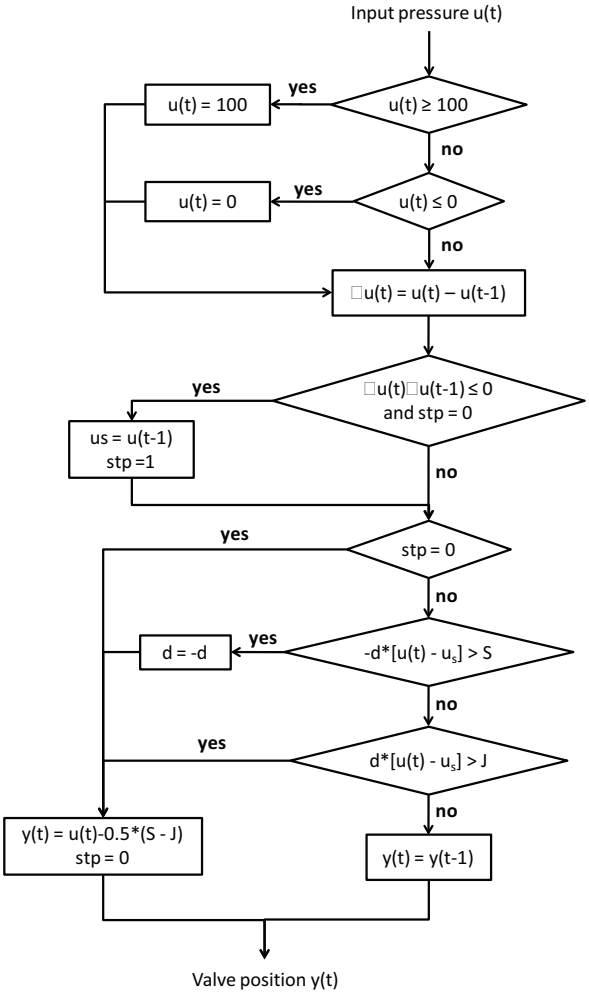


Fig. 2. Flowchart for Kano model.

a second method proposed by Choudhury *et al.* (2008) describes a method based on a grid search. Initially, a grid using several different values of *J* and *S* is built and then based on the process output, the plant model is identified. Based on the mean square error (MSE) between predicted process output and actual output in each grid point, the stiction parameters are estimated.

3 STICTION QUANTIFICATION

This section describes a new method to compute both stiction parameters and plant model, using only normal operating data. Data from process variable (PV) and controller output (OP) are required. Here, only first order with time delay models (FOPTD) will be used. However, the methodology is adequate for second orders, integrating process, among others.

Our approach uses the following assumptions, which are quite similar to the other methods available in the literature:

- The plant model is (locally) linear;
- The loop nonlinearity is caused by the valve;
- The stiction model can be considered a Hammerstein model;

The proposed method computes both plant stiction parameters in a single step, using a global optimization algorithm. This is the first difference between this work and the work proposed by Jelali (2008), where a two step procedure is proposed.

3.1 Optimization problem

The optimization problem to be solved is a non-linear programming type, where the objective function is the mean square error (MSE) of the difference of model output (PV_P) and plant output (PV).

$$MSE = \sum_{i=1}^{n} \frac{\left(PV_i - PV_{P_i}\right)^2}{n}$$

Where *n* is the length of PV.

In this class of problems, the function inside the search space is non-smooth and has some flat areas, where the gradient is zero, or close to this values. Fig 3 illustrates this behavior, using a FOPTD model and a sticky valve. The MSE was computed, varying *J* and *S*. In the process output, white noise is added, with signal-noise ratio equal to 1.

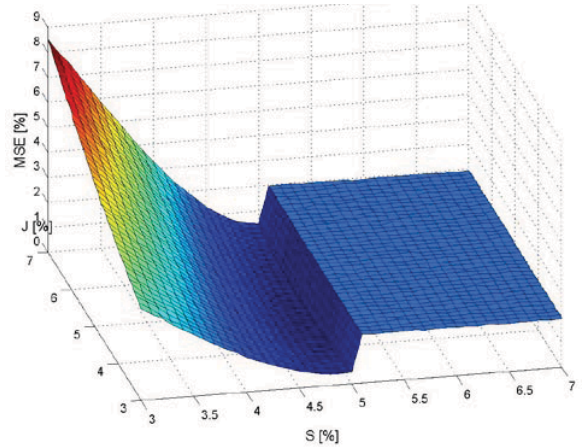


Fig 3: Objective function shape for variable S and J

Fig. 3 clearly shows that this class of function pattern requires global search algorithms; otherwise probably it will stick, depending on the initial guess. One possibility is to use stochastic algorithms, as proposed by Jelali (2008). A second possibility is to use global optimization deterministic algorithms, where the convergence is guaranteed, as proposed in this work.

The optimization problem for a FOPTD model to be solved is:

$$\min_{J,S,K,\tau} (MSE)$$

Where K and τ are the static gain and process time constant, respectively. The time delay is assumed to be known. Several methodologies available in the literature can be used to estimate this loop parameter (Elnaggar *et al.*, 1991, Ahmed *et al.*, 2006, Liang *et al.*, 2003).

The proposed technique can be easily extended to higher order or integrating processes. In this case, the plant model is replaced by an integrating transfer function $\left(\frac{K}{As}\right)$. In this case, the *K* and *A* are estimated by the optimization algorithm.

To allow the industrial application of the proposed method, the computational time should be reasonable. Thus, an efficient global optimization algorithm should be selected. Several optimization methods have been evaluated, and the best obtained by the authors is DIRECT (Finkel and Kelley, 2006). All algorithms are deterministic and deal with bounded decision variables.

4 CASE STUDIES

This section illustrates the applicability of the proposed methodology. Over a thousand simulation systems and a dozen of industrial sticky valves were analyzed and the proposed methodology has shown reliable results, what corroborates its industrial usefulness. Some of these systems will be shown here.

All computations were performed in an Intel Pentium D, 2GHz with 1 GB Ram.

4.1 Simulation case-studies

The objective of this section is to show the applicability of the proposed method in a set of simulation studies. All simulations use a PI controller and a first order plus time delay transfer function. Tab. 1 shows the models used in this work.

Tab. 1: Process and controller models used in the simulation case studies

Parameter	Model
Plant	$G(s) = \frac{1}{\tau s + 1} e^{-3s}$
Controller (PI)	$C(s) = 2\frac{\tau s + 1}{\tau s}$

Here, only twelve cases are shown, where a closed loop system is investigated with variable stiction parameters (*S* and *J*) and different plant time constant. The remaining parameters are maintained constant. Kano's model was used in all cases. The stiction parameters are specified as percentage of input and process variable span (%). Tab. 2 provides the summary of the results obtained by the application of the proposed methodology, where the true plant and valve stiction parameters (τ, *S*, and *J*) were compared with their values obtained by the proposed methodology. (τ_p, S_p, and J_p). The computation time (CPU_t) is also shown. All default settings in the DIRECT algorithm were used, except the maximum number of evaluations of objective functions, which was increased by 1000.

Tab. 2 corroborates the applicability of the proposed method, where the model parameters have deviation less than 3% of

the actual values. These values are comparable with Jelali's simulation cases, where the errors are less than 10%. If the maximum number of evaluations of objective function is increased by 3000, then the deviation reduces by less than 1%, however the CPU time increases to 12 min each.

Tab. 2: Results for process simulations

Case	J (%)	J_p (%)	Error (%)	S (%)	S_p (%)	Error (%)	τ (min)	τ_p (min)	Error (%)	CPU_t (min)
1	2,3	2,3	0,1%	3,0	3,0	0,1%	30,0	30,0	0,0%	4.3
2	2,3	2,3	0,1%	3,0	3,0	0,1%	10,0	10,0	0,0%	4.3
3	3,0	3,0	0,1%	3,0	3,0	0,1%	30,0	30,0	0,0%	4.3
4	3,0	3,0	0,0%	3,0	3,0	0,1%	10,0	10,0	0,0%	4.3
5	3,8	3,8	0,1%	3,0	3,0	0,1%	30,0	30,0	0,0%	4.2
6	3,8	3,7	-1,2%	3,0	3,0	0,0%	10,0	9,9	-1,0%	4.2
7	0,8	0,8	0,6%	1,0	1,0	1,1%	30,0	30,0	0,1%	4.4
8	0,8	0,8	2,9%	1,0	1,0	3,1%	10,0	10,1	1,3%	4.4
9	1,0	1,0	0,0%	1,0	1,0	0,1%	30,0	30,0	0,0%	4.0
10	1,0	1,0	-1,7%	1,0	1,0	-1,2%	10,0	10,0	-0,1%	4.1
11	1,3	1,3	1,3%	1,0	1,0	0,4%	30,0	30,1	0,2%	4.0
12	1,3	1,2	-1,6%	1,0	1,0	-0,9%	10,0	10,0	0,0%	4.3

The second factor also analyzed in this work, was the impact of white noise. Using the same case study of Tab. 1 with $\tau = 20, J = 5$, and $S = 5$, and different level of added white noise to the process variable several optimizations have been performed and the results are summarized in Tab. 3 where SNR is the relationship between Signal-Noise Ratio and the predicted stiction parameters, expressed in percentage of actual value.

Tab 3. White noise impact over the predicted stiction parameters – % change in each parameter

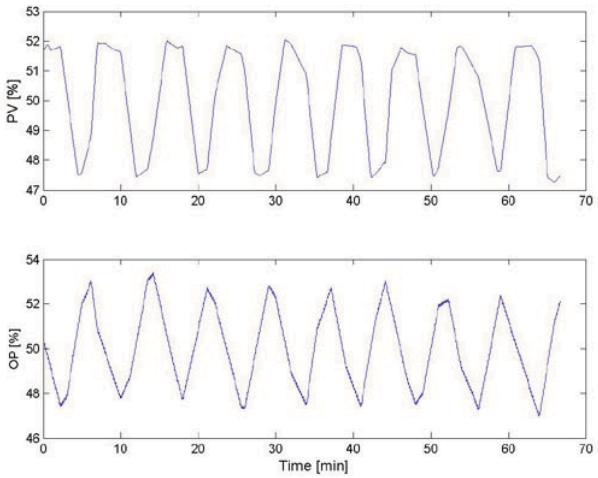
SNR	% S	% J
100	0.09%	-0.23%
50	0.29%	0.87%
5	0.50%	1.23%
0.5	5.9%	25%

As shown in Tab 3, the methodology is not very sensitive to white noise impact. Only when the noise is significant (i.e. SNR = 0.5) the results have been corrupted.

4.2 Industrial case-studies

This section shows some of the industrial application where the proposed methodology was applied. One flow control (case 1) and one pressure control (case 2) with sticky valves, from a Brazilian refinery, are analyzed.

Fig. 4 illustrates the PV and OP for industrial case study 1, where the presence of stiction can be easily seen. The application of the procedure proposed in this work leads to the estimates of $J = 2.6$, $S = 4.0$, and $\tau = 80$ sec. The comparison between the measured and predicted curves is shown in Fig. 5. This comparison shows that the estimated curve is in good agreement with the measured process variable.



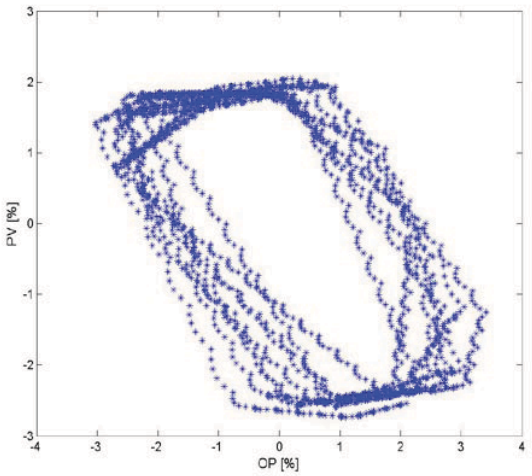


Fig. 4: Data trend for industrial case study 1 – flow control.

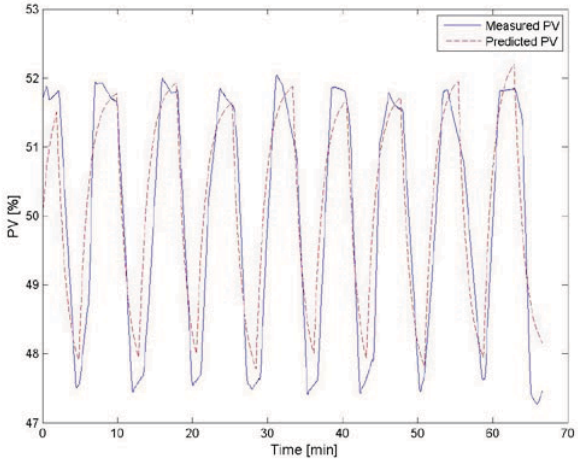


Fig 5: Comparison between measured and predicted PV for industrial case study 1 – flow control.

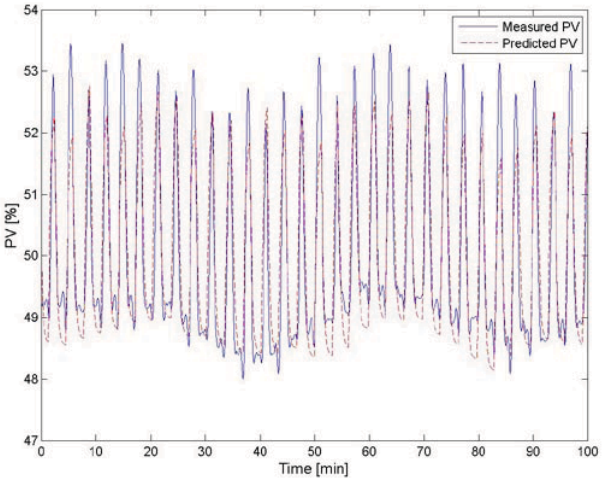
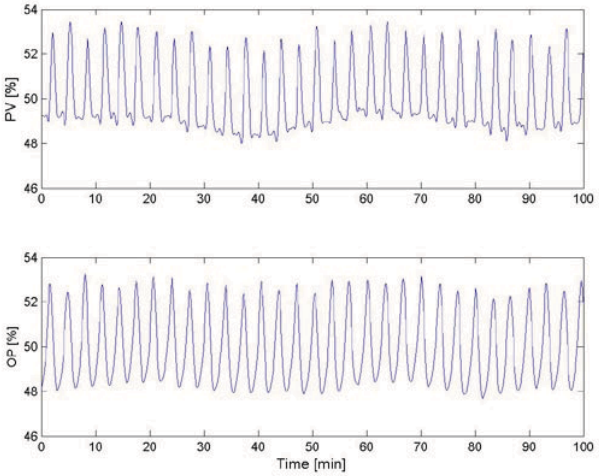


Fig 7 Comparison between measured and predicted PV for industrial case study 2 – pressure control.



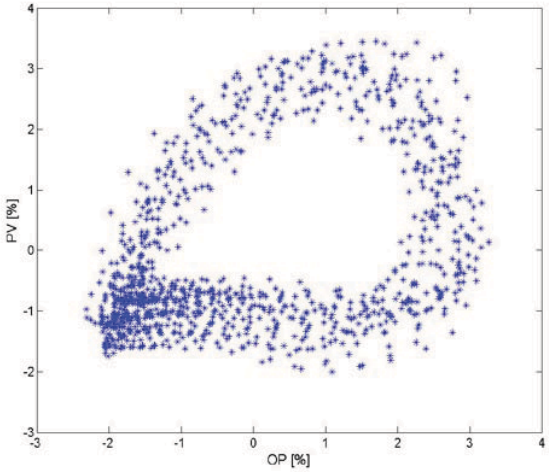


Fig. 6: Data trend for industrial case study 2 – pressure control.

The PV and OP signals for the second industrial sticky valve are shown in Fig. 6. Again, the stiction can be detected by visual inspection of PV versus OP plot, where a parallelogram shape is seen. The proposed estimation algorithm leads to the parameters estimates: $J = 1.6$, $S = 2.9$, and τ=18 sec. The comparison between the measured and predicted curves is shown in Fig. 7.

5 CONCLUSIONS

This work proposes a new method for quantifying valve stiction based on global optimization, using a one-step procedure, where both stiction parameters and plant model are simultaneously quantified, using only process variable (PV) and controller output (OP). The objective function minimized the mean square error between the measured and predicted process output and the optimization algorithm used for this class of problem is called DIRECT (Finkel and Kelley, 2006).

The validity of the method is successfully demonstrated by comparing simulation results, where valves with known stiction parameters were evaluated. Industrial valves were also evaluated, providing very good results. Comparing the actual procedure with the available in the literature, the CPU time is considerably smaller – in this case lower than 5 min against 20 to 30 min – and the quality of the results is comparable – an error lower than 3% against 10%. The industrial applicability of the proposed method has been corroborated by two industrial applications, where reliable results have been obtained.

ACKNOWLEDGMENTS

The authors are very grateful for the grants from CAPES / BRAZIL.

REFERENCES

Ahmed, S., Huang, B. and Shah, S. L. (2006) Parameter and delay estimation of continuous-time models using a linear filter. *Journal of Process Control,* 16**,** 323-331.

Bialkowski, W. L. (1993) Dreams versus reality: A view from both sides of the gap. *Pulp and Paper Canada,* 94**,** 19-27.

Brown, B. P. (1958) Ground Simulator Studies of the Effects of Valve Friction, Stick Friction, Flexibility, and Backlash on Power Control System Quality. IN Aeronautics, N. A. C. F. (Ed.) *Report 1348.* Washington D.C., National Advisory Committee for Aeronautics.

Chen, S. L., Tan, K. K. and Huang, S. (2008) Two-layer binary tree data-driven model for valve stiction. *Industrial and Engineering Chemistry Research,* 47**,** 2842-2848.

Choudhury, M. A. A. S., Jain, M. and Shah, S. L. (2006) Detection and quantification of valve stiction. *Proceedings of the American Control Conference.*

Choudhury, M. A. A. S., Jain, M. and Shah, S. L. (2008) Stiction - definition, modelling, detection and quantification. *Journal of Process Control,* 18**,** 232-243.

Choudhury, M. A. A. S., Thornhill, N. F. and Shah, S. L. (2005) Modelling valve stiction. *Control Engineering Practice,* 13**,** 641-658.

Elnaggar, A., Dumont, G. A. and Elshaefei, A.-L. (1991) Delay estimation using variable regression. *Proceedings of the American Control Conference.*

Finkel, D. E. and Kelley, C. T. (2006) Additive scaling and the DIRECT algorithm. *Journal of Global Optimization,* 36**,** 597-608.

Garcia, C. (2008) Comparison of friction models applied to a control valve. *Control Engineering Practice,* 16**,** 1231-1243.

He, Q. P., Wang, J., Pottmann, M. and Qin, S. J. (2007) A curve fitting method for detecting valve stiction in oscillating control loops. *Industrial and Engineering Chemistry Research,* 46**,** 4549-4560.

Horch, A. (1999) A simple method for detection of stiction in control valves. *Control Engineering Practice,* 7**,** 1221-1231.

Jelali, M. (2008) Estimation of valve stiction in control loops using separable least-squares and global search algorithms. *Journal of Process Control,* 18**,** 632-642.

Kano, M., Maruta, H., Kugemoto, H. and Shimizu, K. (2004) Practical model and detection algorithm for valve stiction. IN Ifac (Ed.) *7th IFAC DYCOPS.* Boston, USA.

Liang, Y., An, D. X., Zhou, D. H. and Pan, Q. (2003) Estimation of time-varying time delay and parameters of a class of jump Markov nonlinear stochastic systems. *Computers & Chemical Engineering,* 27**,** 1761-1778.

Rossi, M. and Scali, C. (2005) A comparison of techniques for automatic detection of stiction: simulation and application to industrial data. *Journal of Process Control,* 15**,** 505-514.

Ruel, M. (2000) Stiction: The hidden menace. *Control Magazine*.

Scali, C. and Ghelardoni, C. (2008) An improved qualitative shape analysis technique for automatic detection of valve stiction in flow control loops. *Control Engineering Practice*.

Singhal, A. and Salsbury, T. I. (2005) A simple method for detecting valve stiction in oscillating control loops. *Journal of Process Control,* 15**,** 371-382.

Stenman, A., Gustafsson, F. and Forsman, K. (2003) A segmentation-based method for detection of stiction in control valves. *International Journal of Adaptive Control and Signal Processing,* 17**,** 625-634.

Yamashita, Y. (2006) An automatic method for detection of valve stiction in process control loops. *Control Engineering Practice,* 14**,** 503-510.

Optimization and Optimal Control

Oral Session

Nonsmooth Optimization of Systems with Varying Structure

Mehmet Yunt, * **Paul I. Barton** **

* *Dept. of Mechanical Engineering, Massachusetts Institute of Technology, Cambridge, MA 02139, USA (e-mail: myunt@mit.edu)*
** *Dept. of Chemical Engineering, Massachusetts Institute of Technology, Cambridge, MA 02139, USA (e-mail: pib@mit.edu)*

Abstract: A novel method based on the generalized gradient and nonsmooth optimization techniques called bundle methods is introduced to optimize the performance of a class of dynamic systems whose governing equations change depending on the values of the parameters, controls and the current state of the system.

Keywords: Nonsmooth Optimization, Generalized Gradient, Bundle Methods, Dynamic Optimization

1. INTRODUCTION

This paper describes a novel method to determine the optimal controls and parameters for a large class of engineering systems with varying structure which have the following characteristics:

(1) The systems evolve according to different ordinary differential equations depending on the values of the states, controls and parameters.
(2) The vector field is unique and continuous.
(3) The values of the continuous states, controls and parameters solely determine the vector field.

Example 1. The liquid level dynamics of a tank with a weir with multiple inlet and outlet flows is

$$\dot{h}(t,\mathbf{p}) = \sum_{i=1}^{n} F_i(t)/A - F_W(h(t,\mathbf{p}),\mathbf{p}) \tag{1}$$

$$F_W(h(t,\mathbf{p}),\mathbf{p}) = \begin{cases} 0 & \text{if } h(t,\mathbf{p}) \le \bar{h} \\ k(h(t,\mathbf{p}) - \bar{h}) & \text{if } h(t,\mathbf{p}) \ge \bar{h} \end{cases}$$

$$\mathbf{p} = (\bar{h}, k, A)^{\mathrm{T}}$$

where h is the liquid level; A is the cross-sectional area of the tank; F_i are volumetric flows; $\bar{h}$ is the weir height; k is an equivalent valve constant for the weir. The governing ordinary differential equations are determined by h and $\bar{h}$. At $h = \bar{h}$, two ordinary differential equations are applicable but the vector field is unique and continuous.

These systems can be expressed in the form,

$$\dot{\mathbf{x}}(t,\mathbf{p}) = \mathbf{f}(\mathbf{x}(t,\mathbf{p}),\mathbf{u}(t),\mathbf{p})$$

where $\mathbf{x}$ represents the continuous-valued states, $\mathbf{p}$ is a finite set of continuous-valued parameters, $\mathbf{u}(t)$ are the bounded controls with possible discontinuities at finitely many points in time. $\mathbf{f}$ is continuous on its domain. The domain of $\mathbf{f}$ is partitioned into finitely many sets, D_k, such that if $(\mathbf{x}(t,\mathbf{p}),\mathbf{u}(t),\mathbf{p}) \in D_k$ then $\mathbf{f}(\mathbf{x}(t,\mathbf{p}),\mathbf{u}(t),\mathbf{p}) = \mathbf{f}_k(\mathbf{x}(t,\mathbf{p}),\mathbf{u}(t),\mathbf{p})$ where $\mathbf{f}_k$ are continuously differentiable with respect to their arguments. As a result of this particular structure $\mathbf{f}$ is piecewise continuously differentiable with respect to its arguments.

In order to formulate a finite dimensional optimization problem, the controls are reformulated as piecewise constant functions of parameters and time with finitely many discontinuities at $t_i,\ i = 1,\ldots,n$ with $t_i < t_{i+1}$, $t_1 = 0$ and $t_n < \infty$. As a result of this reformulation, the dynamics can be represented by a set of equations $\dot{\mathbf{x}}(t,\mathbf{p}) = \mathbf{f}^i(\mathbf{x}(t,\mathbf{p}),\mathbf{p})$ if $t \in (t_i, t_{i+1}]$. Note that each $\mathbf{f}^i$ has a partitioned domain with finitely many partitions, $\{D_k^i,\ k = 1,\ldots,n_i\}$, and corresponding continuously differentiable vector fields $\{\mathbf{f}_k^i\}$.

The mathematical program to be solved is

$$\min_{\mathbf{p}} J(\mathbf{p}) = \sum_{i=1}^{n-1} \int_{t_i}^{t_{i+1}} h(\mathbf{x}(t,\mathbf{p}),\mathbf{p})dt + H(\mathbf{x}(t_n,\mathbf{p}),\mathbf{p}) \tag{2}$$

$$\text{s.t} \sum_{i=1}^{n-1} \int_{t_i}^{t_{i+1}} \mathbf{g}(\mathbf{x}(t,\mathbf{p}),\mathbf{p})dt + \mathbf{G}(\mathbf{x}(t_n,\mathbf{p}),\mathbf{p}) \le \mathbf{0}$$

$$\dot{\mathbf{x}}(t,\mathbf{p}) = \mathbf{f}^i(\mathbf{x}(t,\mathbf{p}),\mathbf{p}),\ \forall t \in (t_i, t_{i+1}], \tag{3}$$

$$\mathbf{x}(0,\mathbf{p}) = \mathbf{f}^0(\mathbf{p}),\ i = 1,\ldots,n-1$$

where H, h, $\mathbf{G}$, $\mathbf{g}$, $\mathbf{f}^0$ are continuously differentiable functions of their arguments that are used to compute values of the objective, values of the constraints and the initial conditions.

Standard numerical methods of dynamic optimization (Betts, 1998) are not applicable to solve (2) because the $\mathbf{f}^i$ are not continuously differentiable. Sensitivity results (Galán et al., 1999) are also not applicable because given $t \in (t_i, t_{i+1}]$, it is not a priori known which $\mathbf{f}_k^i$ govern the dynamics of the system. Furthermore, at $t \in (t_i, t_{i+1}]$ the governing $\mathbf{f}_k^i$ depends on the value of $\mathbf{p}$. The solution of (2) not only determines an optimal $\mathbf{p}$ but prescribes a sequence of $\mathbf{f}_k^i$ in time. This implicit selection complicates the solution process.

The *mixed-integer* (Avraam et al., 1998; Bemporad and Morari, 1999) and the *differential variational inequalities, DVI* (Schumacher, 2004; Pang and Stewart, 2008; Raghunathan et al., 2004) approaches make this selection explicit using transcription. Given $[t_i, t_{i+1}]$, a mesh of time points $\{\tau_j : \ j = 1, 2, \dots n_i, \ \tau_1 = t_i \ \tau_{n_i} = t_{i+1}\}$ is determined a priori. For each τ_j, a variable, $\mathbf{x}_j$, representing the continuous state is created. In order to make the selection of $\mathbf{f}_k^i$ explicit, (3) needs to be replaced with suitable algebraic relationships and additional constraints. The governing dynamics need to be determined in the intervals $[\tau_j, \tau_{j+1}]$. Both approaches replace (3) with a discretization such as the forward Euler method. Both methods introduce additional variables, μ_j^k, at each τ_j for each $\mathbf{f}_k^i$ and approximate the dynamics between time points using the function; $\sum_k \mu_j^k \mathbf{f}_k^i(\mathbf{x}_j, \mathbf{p})$. For example, if the forward Euler discretization is used, an algebraic relationship replacing (3) is $\mathbf{x}_{j+1} - \mathbf{x}_j = (\tau_{j+1} - \tau_j) \sum_{k=1}^{n_i} \mu_j^k \mathbf{f}_k^i(\mathbf{x}_j, \mathbf{p}), \ j = 1, \dots, n_i - 1$. Both approaches replace the integral terms in (2) by appropriate quadratures.

The approaches differ in the way the values of μ_j^k are determined. In the mixed-integer approach, μ_j^k are binary variables. Additional constraints enforce that a single μ_j^k is non-zero at a given τ_j and that this is consistent with parameter, $\mathbf{p}$, and state values, $\mathbf{x}_j$ at τ_j. The final formulation is a mixed-integer nonlinear program. In the DVI approach, μ_j^k are part of the solution of an embedded mathematical program of the states and parameters at each τ_j. These programs are replaced with their equivalent KKT conditions.

Both approaches result in large optimization problems as a result of transcription. Only linear and quadratic formulations of the mixed-integer approach can be solved practically. This restricts the underlying ODEs to be linear and the sets D_k to be polyhedral. The DVI approach requires that the embedded programs are convex to guarantee that the KKT conditions are sufficient to determine an optimal solution. Complementarity constraints in the KKT conditions necessitate special solvers because complementarity conditions violate constraint qualifications which are necessary for ordinary nonlinear programming, NLP, algorithms to work. The required special solvers are not as efficient as usual NLP solvers.

This paper describes a method where the selection of dynamics is not handled explicitly. The method is based on *single-shooting*. In single shooting, the dynamics in (3) are solved by an initial value solver, IVP. In the differentiable case, the IVP solver is also used to solve an auxiliary set of equations to obtain parametric sensitivities. These sensitivities are used to calculate gradient information for numerical optimization methods. There are no additional variables or constraints. The resultant optimization problems do not grow with the number of time points in the mesh nor the size of the possible set of ODEs. In addition, the convexity constraints on the dynamics as in the mixed-integer and DVI approaches can be relaxed.

In order to use single-shooting on (2), which is a nonsmooth program, derivative-like information needs to be obtained. Nonsmoothness also prevents the application of standard nonlinear programming solvers. In order to handle these complications, *Clarke's generalized Jacobian* (Clarke, 1990) is employed in conjunction with a class of nonsmooth optimization methods called *bundle methods* (Kiwiel, 1985; Mäkelä, 2001) to solve (2).

In the remainder, the necessary mathematical background is summarized. The method is then described. The paper ends with an illustrative example and directions of further research.

2. MATHEMATICAL BACKGROUND

$\mathbf{F} : \mathbb{R}^n \to \mathbb{R}^m$ is *locally Lipschitz continuous* at $\mathbf{x}$ if there exists a neighborhood, $N_\epsilon(\mathbf{x})$ and a finite positive constant, K, such that $\|\mathbf{F}(\mathbf{y}) - \mathbf{F}(\mathbf{z})\| \leq K\|\mathbf{y} - \mathbf{z}\| \ \forall \mathbf{z}, \mathbf{y} \in N_\epsilon(\mathbf{x})$. Rademacher's theorem states that locally Lipschitz continuous functions are almost everywhere differentiable on their domain (Rockafellar and Wets, 1998). The locally Lipschitz continuous property is preserved under addition and composition of functions.

Let $\mathbf{F} : \mathbb{R}^n \to \mathbb{R}^m$ be locally Lipschitz continuous at $\mathbf{x}$, then the *generalized Jacobian* (Clarke, 1990) at $\mathbf{x}$ is

$$\partial \mathbf{F}(\mathbf{x}) = \text{co}\{ \lim_{\mathbf{x}_i \to \mathbf{x}} J\mathbf{F}(\mathbf{x}_i) : \ \mathbf{x}_i \notin S \cup T\} \quad (4)$$

where co is the convex hull, $J\mathbf{F}$ denotes the Jacobian of $\mathbf{F}$ where it exists; S is the set of measure zero nondifferentiable points and T is an arbitrary set of measure zero. In words, the generalized Jacobian at $\mathbf{x}$ is the convex hull of all the limits of convergent Jacobian sequences with the Jacobians evaluated at points converging to $\mathbf{x}$. In finite dimensional Euclidean spaces, the *generalized gradient* is the generalized Jacobian when $m = 1$ and the elements of the generalized Jacobian are transposed.

Example 2. $\dot{h}(t, \mathbf{p})$ in (1) is locally Lipschitz continuous with respect to h and the generalized gradient is:

$$\partial_{\bar{h}} \dot{h}(t, \mathbf{p}) = \begin{cases} 0 & \text{if } h(t, \mathbf{p}) < \bar{h} \\ \text{co}[0, k] & \text{if } h(t, \mathbf{p}) = \bar{h} \\ k & \text{if } h(t, \mathbf{p}) > \bar{h}. \end{cases}$$

Chain rules can be derived for generalized gradients and Jacobians. Implicit function theorems can be formulated. Necessary conditions of optimality for mathematical programs with locally Lipschitz continuous functions can be defined in terms of generalized gradients. For numerical methods, if the generalized gradient at a point is known, a direction of descent can be obtained by using the element of minimum norm.

In general, it is not possible to calculate all the elements of the generalized gradient at a point to determine directions of descent. *Bundle Methods* (Kiwiel, 1985; Mäkelä, 2001) use an approximation to the generalized gradient to solve

$$\min_{\mathbf{z} \in Z} f(\mathbf{z}) \text{ s.t: } g_m(\mathbf{z}) \leq 0, \ m = 1, \dots, M$$

where f and g_m are locally Lipschitz continuous functions. Bundle methods require that only an element of $\partial f(\mathbf{z})$ and of each $\partial g_m(\mathbf{z})$ are available. The generalized gradient at an iterate is approximated by the convex

hull of a set of generalized gradients of nearby points called the *bundle*. The element of minimum norm of the approximation is used as the descent direction. Using a specialized line search procedure, either the next iterate is determined or another element is added to the bundle to change the direction of descent.

Bundle methods converge to stationary points satisfying KKT type conditions under a *semismoothness* (Mifflin, 1977; Qi, 1993) assumption on f and g_m. Semismoothness guarantees that the iterative line search algorithm in bundle methods terminates after finite number of iterations. Similar to local Lipschitz continuity, semismoothness is conserved under addition, multiplication and composition. Piecewise continuously differentiable functions, finite convex functions and continuously differentiable functions are all examples of functions that are semismooth and locally Lipschitz continuous.

3. METHOD DESCRIPTION

In this section, the theoretical discussion focuses on the dynamic systems,

$$\begin{aligned}&\dot{\mathbf{x}}(t,\mathbf{p}) = \mathbf{f}^i(\mathbf{x}(t,\mathbf{p}),\mathbf{p}),\ \forall t \in (t_i, t_{i+1}] \\ &\mathbf{x}(t_1,\mathbf{p}) = \mathbf{f}^0(\mathbf{p}),\ \mathbf{p}\in P,\ i = 1\dots n-1\end{aligned} \tag{5}$$

where $\mathbf{f}^i : \mathbb{R}^{n_x}\times\mathbb{R}^{n_p}\to\mathbb{R}^{n_x}$ are piecewise continuously differentiable with respect to their arguments, $\mathbf{f}^0 : \mathbb{R}^{n_p}\to\mathbb{R}^{n_x}$ is continuously differentiable and P is a compact set with non-empty interior. In order to develop a numerical method, the following assumptions are made:

Assumption 1. Equation (5) has a solution on $[t_i, t_{i+1}]$, $i = 1,\dots,n-1$ for each $\mathbf{p}\in P$.

Assumption 2. The domain of each $\mathbf{f}^i$ is partitioned into a finite set of subdomains with nonempty interior, $\{D_k^i,\ k = 1,\dots,n_i\}$ and $D_k^i = \{(\mathbf{v},\mathbf{p}) : d_{k,j}^i(\mathbf{v},\mathbf{p}) \le 0,\ j = 1,\dots,n_{i,k}\}$ where $d_{k,j}^i$ are continuously differentiable. In other words, the partitions have boundaries that can be expressed using continuously differentiable functions and $d_{k,j}^i(\mathbf{v},\mathbf{p}) = 0$ represent continuously differentiable manifolds of dimension $n_x\times n_p - 1$. There exists a corresponding set of $\{\mathbf{f}_k^i\}$ such that $\mathbf{f}^i(\mathbf{x}(t,\mathbf{p}),\mathbf{p}) = \mathbf{f}_k^i(\mathbf{x}(t,\mathbf{p}),\mathbf{p})$ if $(\mathbf{x}(t,\mathbf{p}),\mathbf{p})\in D_k^i$ and $\mathbf{f}_k^i$ are continuously differentiable with respect to their arguments.

$\mathbf{x}(t,\mathbf{p})$ is a locally Lipschitz continuous (Coddington and Levinson, 1955) and semismooth function of $\mathbf{p}$ (Pang and Stewart, 2009) as a result of continuity and piecewise continuous differentiability of $\mathbf{f}^i$. The constraints and objective of (2) are composite functions of semismooth and locally Lipschitz functions and $\mathbf{x}(t,\mathbf{p})$. As a result, the constraints and objective of (2) are locally Lipschitz and semismooth functions.

In order to calculate the necessary generalized gradient information for optimization, an element of the generalized gradient of the states with respect to $\mathbf{p}$, $\partial_{\mathbf{p}}\mathbf{x}(t,\mathbf{p})$, is required. The next theorem provides a sufficient condition to detect points where an element can be calculated. It can be deduced from Theorem 7.4.1 in Clarke (1990) using appropriate chain rules and the definition of the Jacobian.

Theorem 1. Let $\mathbf{x}(t,\bar{\mathbf{p}})$ be a solution of (5). If the set-valued mapping, $\partial_{(\mathbf{x},\mathbf{p})}\mathbf{f}_{\mathbf{p}}^i(\mathbf{x}(t,\bar{\mathbf{p}}),\bar{\mathbf{p}})$ is a singleton for almost all $t\in(t_1,t_n]$, then for each $i = 1,\dots,n-1$, there exist unique solutions to the matrix differential inclusions:

$$\begin{aligned}&\dot{Y}_{\mathbf{p}}(t)\in\partial\mathbf{f}_{\mathbf{x}}^i(\mathbf{x}(t,\bar{\mathbf{p}}),\bar{\mathbf{p}})Y_{\mathbf{p}}(t) + \partial\mathbf{f}_{\mathbf{p}}^i(\mathbf{x}(t,\bar{\mathbf{p}}),\bar{\mathbf{p}}), \\ &\qquad\qquad\qquad\qquad \forall t\in(t_i,t_{i+1}] \\ &\dot{Y}_{\mathbf{x}}(t)\in\partial\mathbf{f}_{\mathbf{x}}^i(\mathbf{x}(t,\bar{\mathbf{p}}),\bar{\mathbf{p}})Y_{\mathbf{x}}(t), \forall t\in(t_i,t_{i+1}] \\ &Y_{\mathbf{p}}(t_1) = 0,\ Y_{\mathbf{x}}(t_1) = I.\end{aligned}$$

The following relations hold for all $t\in(t_i,t_{i+1}]$ except on set of measure zero,

$$\dot{Y}_{\mathbf{p}}(t) = J\mathbf{f}_{\mathbf{x}}^i(\mathbf{x}(t,\bar{\mathbf{p}}),\bar{\mathbf{p}})Y_{\mathbf{p}}(t) + J\mathbf{f}_{\mathbf{p}}^i(\mathbf{x}(t,\bar{\mathbf{p}}),\bar{\mathbf{p}}) \tag{6}$$

$$\dot{Y}_{\mathbf{x}}(t) = J\mathbf{f}_{\mathbf{x}}^i(\mathbf{x}(t,\bar{\mathbf{p}}),\bar{\mathbf{p}})Y_{\mathbf{x}}(t) \tag{7}$$

where $J\mathbf{f}_{\mathbf{x}}^i$ and $J\mathbf{f}_{\mathbf{p}}^i$ are the partial derivatives of $\mathbf{f}^i$ with respect to $\mathbf{x}$ and $\mathbf{p}$ respectively.

Finally, $\partial_{\mathbf{p}}\mathbf{x}(t_n,\bar{\mathbf{p}}) = Y_{\mathbf{x}}(t_n)J\mathbf{f}_0(\bar{\mathbf{p}}) + Y_{\mathbf{p}}(t_n)$.

Definition 1. A trajectory, $\mathbf{x}(t,\mathbf{p})$, is called *singleton* if $\partial\mathbf{f}_{(\mathbf{x},\mathbf{p})}^i(\mathbf{x}(t,\mathbf{p}),\mathbf{p})$ is a singleton for almost all $t\in[t_i,t_{i+1}]$. Otherwise, it is called *non-singleton*.

Note that due to assumption (2) and the piecewise continuous differentiable nature of $\mathbf{f}^i$, the points where $\partial\mathbf{f}_{(\mathbf{x},\mathbf{p})}^i(\mathbf{x}(t,\mathbf{p}),\mathbf{p})$ may not be a singleton are where $d_{k,j}^i(\mathbf{x}(t,\mathbf{p}),\mathbf{p}) = 0$ holds for some k and j. Trajectories that are not singleton have arcs that lie on the surfaces defined by $d_{k,j}^i(\mathbf{x}(t,\mathbf{p}),\mathbf{p}) = 0$.

The next theorem is a result on the occurrence of non-singleton trajectories for autonomous systems with piecewise continuously differentiable vector fields.

Theorem 2. Consider the dynamic system

$$\dot{\mathbf{x}}(t) = \mathbf{f}(\mathbf{x}(t)),\ t\in(0,t_f],\ \mathbf{x}(0) = \mathbf{x}_0,\ \mathbf{x}_0\in X_0\subset D \tag{8}$$

where X_0 is an open set in $\mathbb{R}^{n_x}$, D is the bounded domain where all solutions with initial conditions in X_0 remain for $t\in(0,t_f]$. Let $\mathbf{f} : \mathbb{R}^{n_x}\to\mathbb{R}^{n_x}$ be piecewise continuously differentiable on D. Let assumptions (2) hold where all $d_{k,j}^i$ are continuously differentiable functions of $\mathbf{x}$. Then the set of initial conditions producing non-singleton trajectories is a measure zero subset of X_0.

Proof. Since $\mathbf{f}$ is locally Lipschitz continuous on D, the set of points where it is not differentiable is a set of measure zero in D. In addition, due to piecewise continuous differentiability, the generalized Jacobian is a singleton where $\mathbf{f}$ is differentiable. The only points where the generalized Jacobian is not a singleton are on the boundaries of the subdomains which constitute a set of measure zero in D. The solutions of (8) are unique due to the Lipschitz continuous property of $\mathbf{f}$ on D. Due to the autonomous nature of the dynamics, no two solutions intersect for any $t\in(0,t_f]$. Now consider trajectories that pass through boundary points. Since points on the boundaries are a set of measure zero in D, the set of initial conditions producing these trajectories are a set of measure zero in D. Since the set of initial conditions that produce non-singleton trajectories is a subset of the set of initial conditions that produce trajectories passing through boundary points, the set of initial conditions producing non-singleton trajectories is a set of measure

zero in D. Finally the intersection of X_0 which is open in $\mathbb{R}^{n_x}$ with this set of initial conditions is a set of measure zero.

If the initial conditions of (8) are functions of a set of parameters, then the result of the previous theorem may not hold. It is possible that the functions always map the parameters to initial conditions resulting in non-singleton trajectories. Sufficient conditions to determine functions that map parameters to sets of initial conditions for which theorem (2) holds is under investigation.

In the remainder of the paper, the following statement is assumed to hold.

Assumption 3. Let $\tilde{P}$ the set of parameters such that $\mathbf{x}(t,\mathbf{p})$ with $\mathbf{p} \in \tilde{P}$ is not a singleton trajectory. The set $\tilde{P}$ has measure zero in P.

The equations (5), (6) and (7) have to be solved numerically. The solution of (5) poses no problems and can be accomplished with an ordinary IVP solver because $\mathbf{f}^i$ are Lipschitz continuous vector field on their domains. On the other hand, solutions of (6) and (7) require further discussion. Since $\mathbf{f}^i$ are only piecewise continuously differentiable, $J_{\mathbf{p}}\mathbf{f}^i$ and $J_{\mathbf{x}}\mathbf{f}^i$ are not continuous on the domain of $\mathbf{f}^i$. As a result (6) and (7) are differential equations with discontinuities in time. These discontinuities need to be detected for an efficient and correct solution. By virtue of assumption (2), the points where $\mathbf{f}^i$ are not differentiable are on the boundaries of the partitions and satisfy $d^i_{k,j}(\mathbf{x}(t,\mathbf{p}),\mathbf{p}) = 0$ for some values of k and j. Event detection algorithms (Park and Barton, 1996) are used to determine t^* such that $d^i_{k,j}(\mathbf{x}(t^*,\mathbf{p}),\mathbf{p}) = 0$. The event detection algorithm tracks the signs of $d^i_{k,j}(\mathbf{x}(t,\mathbf{p}),\mathbf{p})$ for each i and j. A sign change implies that the state trajectory, $\mathbf{x}(t^*,\mathbf{p})$ crossed a boundary of discontinuity. The event detection algorithm finds the earliest time when the boundary crossing occurred. The integration of (6) and (7) are restarted at t^*.

Event detection algorithms are also used to detect trajectories that are non-singleton. Once t^* is detected where $d^i_{k,j}(\mathbf{x}(t^*,\mathbf{p}),\mathbf{p}) = 0$ for some k and j, $\dot{d}^i_{k,j}(\mathbf{x}(t^*,\mathbf{p}),\mathbf{p})$ is checked. If $\dot{d}^i_{k,j}(\mathbf{x}(t^*,\mathbf{p}),\mathbf{p}) = 0$ for some k and j, this implies that the state trajectory may not leave the surface defined by $d^i_{k,j}(\mathbf{x}(t^*,\mathbf{p}),\mathbf{p}) = 0$ resulting in a trajectory that is possibly non-singleton. Integration is continued until $t + \delta t$ where δt is a small quantity. If there exists any k and j such that $d^i_{k,j}(\mathbf{x}(t^*,\mathbf{p}),\mathbf{p}) = 0$ and $d^i_{k,j}(\mathbf{x}(t^*+\delta t,\mathbf{p}),\mathbf{p}) = 0$, the trajectory is considered to be not a singleton trajectory.

In case $\bar{\mathbf{p}}$ does not correspond to a singleton trajectory, the definition of the generalized Jacobian (4) can be used to approximate an element of the generalized Jacobian. Random parameter values in an ϵ neighborhood, $N_\epsilon(\bar{\mathbf{p}})$, can be used to find a nearby singleton trajectory and calculate an approximate generalized Jacobian.

The necessary generalized gradient information for the objective and constraint functions of (2) is obtained by applying the chain rules for generalized gradients once an element or an approximation of $\partial_{\mathbf{p}}\mathbf{x}(t,\mathbf{p})$ is calculated. The calculated generalized gradient information is used in conjunction with a bundle method to obtain a stationary point of (2).

4. ILLUSTRATIVE EXAMPLE

In this section, a modified version of the cell cycle specific chemotherapy model introduced in Pannetta and Adam (1995) is used to determine an optimal chemotherapy drug schedule. The dynamics,

$$\alpha = a - m - n$$

$$\dot{P} = \alpha P + bQ - F_A(v_A, P) \tag{9}$$

$$F_A(v_A,P) = \begin{cases} 0 & \text{if } v_A - \bar{v}_A \le 0 \\ k_A(v_A - \bar{v}_A)P & \text{if } v_A - \bar{v}_A \ge 0 \end{cases}$$

$$\dot{Q} = mP - bQ - F_B(v_B, Q) \tag{10}$$

$$F_B(v_B,Q) = \begin{cases} 0 & \text{if } v_B - \bar{v}_B \le 0 \\ k_B(v_B - \bar{v}_B)Q & \text{if } v_B - \bar{v}_B \ge 0 \end{cases}$$

$$\dot{Y} = \sigma Y(1 - Y/K) - k_A v_A Y - k_B v_B Y \tag{11}$$

$$\dot{v}_A = u_A - \gamma_A v_A \tag{12}$$

$$\dot{v}_B = u_B - \gamma_B v_B \tag{13}$$

represents the behavior of tumor cells and healthy cells in human tissue under chemotherapy. The tissue comprises healthy cells, Y, proliferating tumor cells, P, and quiescent tumor cells, Q. Chemotherapy comprises two drugs; A and B. u_A and u_B are the chemotherapy drug schedules. v_A and v_B are the exponentially decaying drug concentrations in the tissue. Tumor cells develop resistance to drugs. As a result, drugs are effective against the tumor cells only if their concentrations in the tissue are above $\bar{v}_A$ and $\bar{v}_B$. A fraction, n, of proliferating cells die of natural causes and and a fraction, m, of proliferating cells become quiescent cells. The increase in proliferating cell population by cell division is represented as another fraction, a, of the proliferating cell population. In addition, a fraction, b of quiescent cells become proliferating cells. The tumor cell dynamics are in (9) and (10).

A logistic equation (11) governs the healthy cell population to ensure that the number of healthy cells does not exceed the carrying capacity, K. Numerical values for the various parameters are displayed in Table 1. Most of the values are obtained from (Dua et al., 2008) where cell cycle specific chemotherapy with a single drug and without drug resistance is considered. Note that $[D]$ is a unit of drug concentration. The program,

$$\min_{\mathbf{u}_A,\mathbf{u}_B} \; P(t_f,\mathbf{u}_A,\mathbf{u}_B) + Q(t_f,\mathbf{u}_A,\mathbf{u}_B) \tag{14}$$

$$\begin{aligned} \text{s.t: } & Y(t_f,\mathbf{u}_A,\mathbf{u}_B) \ge Y_{\min} \\ & u_{\min} \le u_{A,j} \le u_{\max},\ j = 1,\ldots,n_f, \\ & u_{\min} \le u_{B,j} \le u_{\max},\ j = 1,\ldots,n_f, \\ & P(1,\mathbf{u}_A,\mathbf{u}_B) = P_0,\ Q(1,\mathbf{u}_A,\mathbf{u}_B) = Q_0, \\ & Y(1,\mathbf{u}_A,\mathbf{u}_B) = Y_0,\ v_A(1,\mathbf{u}_A,\mathbf{u}_B) = 0, \\ & v_B(1,\mathbf{u}_A,\mathbf{u}_B) = 0 \end{aligned} \tag{15}$$

where $\mathbf{u}_A = \{u_{A,j}\}$ and $\mathbf{u}_B = \{u_{B,j}\}$ are the set of daily drug doses for an n_f-day treatment, is solved to minimize the tumor cell population without totally destroying the healthy cell population. The numerical values used are in Table 2.

DSL48SE is the IVP solver (Tolsma, 2001; Tolsma and Barton, 2002; Feehery et al., 1997) used to integrate the dynamics and the corresponding auxiliary equations to obtain an element of the generalized Jacobian. The event detection algorithm of DSL48SE (Park and Barton, 1996) is used to detect non-singleton trajectories. The necessary Jacobians for the auxiliary system of equations are obtained using automatic differentiation algorithms implemented in DAEPACK (Tolsma and Barton, 2000). The differential equations are integrated with an absolute tolerance of 1×10^{-7} and a relative tolerance of 1×10^{-9}.

A modified proximal bundle method based on the algorithm in (Lukšan and Vlček, 2001) is used to solve (14). A penalty approach to handle (15) is used because the algorithm in (Lukšan and Vlček, 2001) is handles only linear constraints on the decision variables. The objective of (14) is augmented with (15) to obtain

$$J(\mathbf{u}_A, \mathbf{u}_B) = P(t_f, \mathbf{u}_A, \mathbf{u}_B) + Q(t_f, \mathbf{u}_A, \mathbf{u}_B) + \mu_k \max(Y_{\min} - Y(t_f, \mathbf{u}_A, \mathbf{u}_B), 0)$$

where μ_k is the penalty parameter. The modified program is successively solved three times with increasing penalty parameter to an optimality tolerance of 1×10^{-5}. The solution of the preceding programs are used as the initial guesses for the following programs. For the first program, the drug schedules are assigned random values between 0.0 and 5.0. The penalty parameter values are 5000, 25000 and 125000.

The cell population numbers at the beginning and end of the treatment are in Table 3. The tumor cell population is reduced to one percent of its initial size. The drug schedules are shown in Figure 1 and Figure 2. The preference to use drug B is clearly seen. The effects of the drugs are proportional to the corresponding cell populations. Therefore using drug B results in more effective treatment as the population of quiescent cells is greater than that of proliferating cells. In addition, the ratio of tumor cells killed to the ratio of healthy cells killed per unit drug concentration is larger for drug B.

The drug B schedule has four distinctive phases. The initial four-day treatment reverses the increase in the tumor cell population by using drug B as much as possible. In the next week, the drug B concentration is allowed to decay to a tolerable level for the patient. The treatment until the last three days keeps the drug B concentration at that tolerable level. In the last days of the treatment, the drug dose is increased to kill the maximum number of tumor cells. This spike in the drug concentration shows its effect on the healthy cell population after the treatment is over and does not affect (15) significantly.

5. CONCLUSION

In this document, a novel method to optimize the performance of a class of systems with varying structure has been introduced. The theoretical basis and an initial implementation has been described. An illustrative numerical example has been presented.

The implementation of the optimization method will be streamlined in the future. There are different variants of

Table 1. Parameters of equations (9)-(13)

a	0.500 day^{-1}	$\bar{v}_A$	10.000 [D]
m	0.218 day^{-1}	$\bar{v}_B$	10.000 [D]
n	0.477 day^{-1}	k_A	8.400×10^{-3} day$^{-1}[D]^{-1}$
b	0.100 day^{-1}	k_B	8.400×10^{-3} day$^{-1}[D]^{-1}$
σ	0.100 day^{-1}	K	$10000M$ cells
γ_A	0.100 day^{-1}	γ_B	0.100 day^{-1}

Table 2. Parameters of the mathematical program (14)

t_f	31 days	$Y_{\min}$	$100M$ cells
n_f	30	Y_0	$10000M$ cells
$u_{\max}$	20.00 $[D]$day^{-1}	Q_0	$8.00 \times 10^5 M$ cells
$u_{\min}$	0.00 $[D]$day^{-1}	P_0	$2.00 \times 10^5 M$ cells

Table 3. Cell Populations at the beginning and end of treatment

	Beginning of Treatment	End of Treatment
Y	$10000M$ cells	$100M$ cells
Q	$8.00 \times 10^5 M$ cells	$4.80 \times 10^4 M$ cells
P	$2.00 \times 10^5 M$ cells	$4.60 \times 10^4 M$ cells

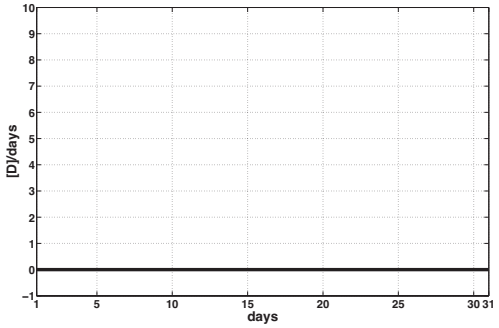


Fig. 1. Drug A Schedule

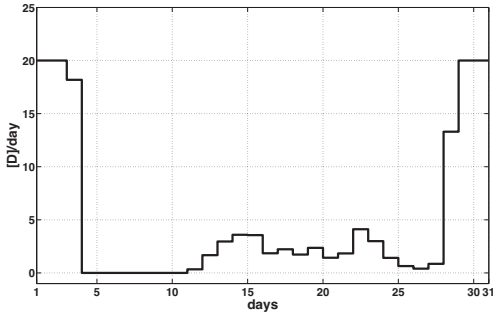


Fig. 2. Drug B schedule

bundle methods available (Mäkelä, 2001). These variants will be investigated in addition to different methods to handle nonlinear constraints. In this paper, a penalty approach is described. Improvements to this approach will be considered and the improvement function approach will be investigated (Mäkelä, 2001).

The performance of the proposed algorithm will be compared to the performance of the available transcription-based algorithms. It is expected that the method introduced in this document can be advantageous in problems with a large number states and candidate dynamics.

REFERENCES

Avraam, M.P., Shah, N., and Pantelides, C. (1998). Modelling and Optimisation of General Hybrid Systems in the Continuous Time Domain. *Computers and Chemical Engineering*, 22(Suppl), S221–S228.

Bemporad, A. and Morari, M. (1999). Control of systems integrating logic, dynamics, and constraints. *Automatica*, 35, 407–427.

Betts, J.T. (1998). Survey of numerical methods for trajectory optimization. *Journal of Guidance Control and Dynamics*, 21(2), 193–207.

Clarke, F.H. (1990). *Optimization and Nonsmooth Analysis*. Number 5 in Classics in Applied Mathematics. SIAM.

Coddington, E.A. and Levinson, N. (1955). *Theory of Ordinary Differential Equations*. McGraw Hill Co., Inc., New York.

Dua, P., Dua, V., and Pistikopoulos, E.N. (2008). Optimal delivery of chemotherapeutic agents in cancer. *Computers & Chemical Engineering*, 32(1-2), 99–107.

Feehery, W.F., Tolsma, J.E., and Barton, P.I. (1997). Efficient sensitivity analysis of large-scale differential-algebraic systems. *Applied Numerical Mathematics*, 25(1), 41–54.

Galán, S., Feehery, W.F., and Barton, P.I. (1999). Parametric sensitivity functions for hybrid / discrete / continuous systems. *Applied Numerical Mathematics*, 31, 17–47.

Kiwiel, K.C. (1985). *Methods of Descent for Nondifferentiable Optimization*, volume 1133 of *Lecture Notes in Mathematics*. Springer-Verlag.

Lukšan, L. and Vlček, J. (2001). Algorithm 811: NDA: Algorithms for nondifferentiable optimization. *ACM Transactions on Mathematical Software*, 27(2), 193–213.

Mäkelä, M.M. (2001). Survey of Bundle Methods for Nonsmooth Optimization. *Optimization Methods and Software*, 17(1), 1 –29.

Mifflin, R. (1977). Semismooth and Semiconvex Functions in Constrained Optimization. *SIAM Journal of Control and Optimization*, 15(6), 959–972.

Pang, J.S. and Stewart, D.E. (2008). Differential variational inequalities. *Mathematical Programming*, 113(2), 345–424.

Pang, J.S. and Stewart, D.E. (2009). Solution dependence on initial conditions in differential variational inequalities. *Mathematical Programming*, 116(1-2), 429–460.

Pannetta, J. and Adam, J. (1995). A mathematical model of cycle-specific chemotherapy. *Mathematical and Computer Modelling*, 22(2), 67–82.

Park, T. and Barton, P.I. (1996). State event location in differential-algebraic models. *ACM Trans. Model. Comput. Simul.*, 6(2), 137–165.

Qi, L. (1993). A nonsmooth version of Newton's method. *Mathematical Programming*, 58, 353–367.

Raghunathan, A., Diaz, M., and Biegler, L. (2004). An MPEC formulation for dynamic optimization of distillation operations. *Computers & Chemical Engineering*, 28(10), 2037–2052.

Rockafellar, R.T. and Wets, R.J.B. (1998). *Variational Analysis*. Number 317 in Grundlehren der mathematischen Wissenschaften. Springer.

Schumacher, J.M. (2004). Complementarity systems in optimization. *Mathematical Programming*, 101(1), 263 – 295.

Tolsma, J. and Barton, P.I. (2000). DAEPACK: an open modeling environment for legacy models. *Industrial & Engineering Chemistry Research*, 39(6), 1826–1839. (http://yoric.mit.edu/daepack/daepack.html).

Tolsma, J.E. (2001). DSL48SE manual (version 1.0). Technical report, Process Systems Engineering Laboratory, Department of Chemical Engineering, Massachusetts Institute of Technology. (http://yoric.mit.edu / daepack / download / Manuals_Pres / dsl48se.ps).

Tolsma, J.E. and Barton, P.I. (2002). Hidden discontinuities and parametric sensitivity analysis. *SIAM Journal on Scientific Computing*, 23(6), 1861–1874.

Real-time Optimization with Estimation of Experimental Gradients

A. Marchetti * **B. Chachuat** ** **D. Bonvin** *

* *Laboratoire d'Automatique, École Polytechnique Fédérale de Lausanne (EPFL), Station 9, CH-1015 Lausanne, Switzerland*
** *Department of Chemical Engineering, McMaster University, 1280 Main Street West, Hamilton, ON L8S 4LT, Canada*

Abstract: For good performance in practice, real-time optimization schemes need to be able to deal with the inevitable plant-model mismatch problem. Unlike the two-step schemes combining parameter estimation and optimization, the modifier-adaptation approach uses experimental gradient information and does not require the model parameters to be estimated on-line. The dual modifier-adaptation approach presented in this paper drives the process towards optimality, while paying attention to the accuracy of the estimated gradients. The gradients are estimated from successive operating points generated by the optimization algorithm. The novelty lies in the development of an upper bound on the norm of the gradient errors, which is used as a constraint when determining the next operating point. The proposed approach is demonstrated in simulation via the real-time optimization of a continuous reactor.

Keywords: Real-time optimization, estimation of experimental gradients, modifier adaptation.

1. INTRODUCTION

Real-time optimization (RTO) of continuous plants aims at improving some steady-state performance index (Marlin and Hrymak [1997]). Since the majority of RTO schemes uses a model of the plant, reaching optimal performance in the presence of plant-model mismatch is a difficult task, which necessitates adaptation based on measured information. Chachuat et al. [2009] proposed a three-way classification of RTO schemes. One class includes the so-called modifier-adaptation approach (Marchetti et al. [2009]), whereby appropriate terms are added to the optimization problem and identified so that the KKT conditions of the model match those of the plant. In this context, the modifier-adaptation approach requires to be able to estimate on-line the experimental gradients, i.e., the derivatives of the plant outputs with respect to the inputs. This paper investigates the estimation of experimental gradients and their use in modifier-adaptation schemes.

A comparison of different approaches for on-line gradient estimation is given in Mansour and Ellis [2003]. Finite-difference techniques can be used to estimate the gradients experimentally. The most straightforward approach consists in perturbating each input individually around the current operating point to get an estimate of the corresponding gradient elements. This is the case, e.g., when forward finite differencing (FFD) is applied at each RTO iteration. An alternative approach, which was introduced in the ISOPE (Integrated System Optimization and Parameter Estimation) literature under the name *dual ISOPE*, is to estimate the gradients based on the current and past operating points (Brdyś and Tatjewski [1994, 2005]). The key issue therein is the ability to estimate the experimental gradients reliably while updating the operating point. Indeed, there are two conflicting objectives: the "primal objective" consists in solving the optimization problem, while the "dual objective" aims at estimating accurate gradients. These conflicting tasks can be accommodated by adding a constraint in the optimization problem so as to ensure sufficiently rich information in the measurements and guarantee gradient accuracy. Brdyś and Tatjewski [1994, 2005] proposed a constraint that prevents ill-conditioning in gradient computation. The present paper goes further and investigates the two main sources of errors, namely the error introduced by numerical approximation of a derivative (truncation error) and measurement noise. A constraint that enforces an upper bound on the gradient error norm is proposed. Since the constraint for ensuring sufficient information might compromise optimality in the vicinity of the optimum, Gao and Engell [2005] suggested using the ill-conditioning measure not to constrain the optimization problem but rather to determine whether an additional input perturbation is needed. Note that such a scheme could also be used in the context of the dual-modifier approach proposed here.

The paper is organized as follows. Section 2 formulates the optimization problem. The modifier-adaptation scheme is reviewed in Section 3. Analysis of the errors in the gradient estimates obtained from past operating points is carried out in Section 4. Based on this analysis, Section 5 proposes a norm-based constraint, which is incorporated into the dual modifier-adaptation algorithm presented in Section 6. The approach is illustrated via the reactor of the Williams-Otto plant in Section 7, and Section 8 concludes the paper and presents directions for future work.

2. PROBLEM FORMULATION

For the sake of simplicity, an unconstrained optimization problem is considered throughout. This way, only the

gradient of the objective function needs to be estimated, while in the constrained case, the constraint gradients would need to be evaluated as well. A possible way of tackling constrained optimization problems will be sketched in the conclusion section.

The unconstrained optimization problem reads:

$$\min_{\mathbf{u}} \quad \Phi_p(\mathbf{u}) := \phi(\mathbf{u}, \mathbf{y}_p(\mathbf{u})) \tag{1}$$

where $\mathbf{u} \in \mathbb{R}^{n_u}$ are the decision (or input) variables, $\mathbf{y}_p \in \mathbb{R}^{n_y}$ are the measured (or output) variables, and $\phi : \mathbb{R}^{n_u} \times \mathbb{R}^{n_y} \to \mathbb{R}$ is the scalar cost function to be minimized. The notation $(\cdot)_p$ will be used for the variables that are associated with the plant. Also, it is assumed that $\phi(\mathbf{u}, \mathbf{y}_p)$ is a known function of $\mathbf{u}$ and $\mathbf{y}_p$. On the other hand, the steady-state input-output mapping of the plant, $\mathbf{y}_p(\mathbf{u})$, is typically unknown, and only the approximate model $f(\mathbf{u}, \mathbf{y}, \boldsymbol{\theta}) = \mathbf{0}$ is available, where $\boldsymbol{\theta} \in \mathbb{R}^{n_\theta}$ is the set of model parameters. Assuming that the model outputs $\mathbf{y}$ can be expressed explicitly as functions of $\mathbf{u}$ and $\boldsymbol{\theta}$, the cost function predicted by the model becomes $\Phi(\mathbf{u}, \boldsymbol{\theta}) := \phi(\mathbf{u}, \mathbf{y}(\mathbf{u}, \boldsymbol{\theta}))$.

It is furthermore assumed that the decision variables $\mathbf{u}$ are of the same order of magnitude, which can be achieved via scaling. For example, if the decision variable u_i remains within the interval $[u_{i,\mathrm{a}}, u_{i,\mathrm{b}}]$, it can be scaled as $u_i^{\mathrm{scaled}} = (u_i - u_{i,\mathrm{a}})/(u_{i,\mathrm{b}} - u_{i,\mathrm{a}})$. For notional simplicity, the superscript indicating a scaled variable will be omitted in the sequel.

3. MODIFIER-ADAPTATION SCHEME

In the modifier-adaptation scheme, a gradient-correction term is added to the cost function of the model-based optimization problem (Marchetti et al. [2009]). At the kth iteration, the next input $\mathbf{u}_{k+1}$ is obtained as:

$$\mathbf{u}_{k+1} = \arg\min_{\mathbf{u}} \Phi_m(\mathbf{u}, \boldsymbol{\theta}) := \Phi(\mathbf{u}, \boldsymbol{\theta}) + \boldsymbol{\lambda}_k^{\mathsf{T}} \mathbf{u} \tag{2}$$

where $\boldsymbol{\lambda}_k$ is the cost-gradient modifier at the kth iteration. This modifier is adapted at each iteration based on the difference between the gradient of the plant and that predicted by the model. For example, upon implementation of a first-order exponential filter, the gradient modifier is calculated as:

$$\boldsymbol{\lambda}_k^{\mathsf{T}} = (1-d)\boldsymbol{\lambda}_{k-1}^{\mathsf{T}} + d\left[\frac{\partial \Phi_p}{\partial \mathbf{u}}(\mathbf{u}_k) - \frac{\partial \Phi}{\partial \mathbf{u}}(\mathbf{u}_k, \boldsymbol{\theta})\right] \tag{3}$$

with the filter gain $d \in (0,\ 1]$. Computation of the modifier $\boldsymbol{\lambda}_k$ requires the knowledge of the plant gradient $\frac{\partial \Phi_p}{\partial \mathbf{u}}(\mathbf{u}_k)$.

An appealing property of the modifier-adaptation scheme is that, upon convergence and in the absence of noise, the optimum $\mathbf{u}_\infty$ for the modified model-based optimization problem (2) satisfies the first-order necessary conditions of optimality of the optimization problem (1) (Marchetti et al. [2009]). Note that this is the case despite plant-model mismatch. Note also that the need to match the plant outputs $\mathbf{y}_p(\mathbf{u})$ by means of a parameter estimation problem, as this is the case for the ISOPE modifier (Roberts [1979]), is removed. However, the downside of modifier adaptation lies in the need to estimate the experimental gradient $\frac{\partial \Phi_p}{\partial \mathbf{u}}$.

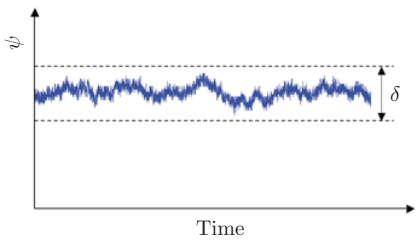


Fig. 1. Noisy cost function at steady state.

4. EXPERIMENTAL GRADIENT COMPUTED FROM PAST OPERATING POINTS

It is assumed that the cost function can be evaluated from the noisy output measurements as follows:

$$\psi(\mathbf{u}) = \phi(\mathbf{u}, \mathbf{y}_p(\mathbf{u}) + \boldsymbol{\nu}) = \Phi_p(\mathbf{u}) + v \tag{4}$$

where $\boldsymbol{\nu}$ is the measurement noise on the outputs and v the induced noise in the cost estimates. Note that, even if $\boldsymbol{\nu}$ is zero mean, v might have a nonzero mean if the function $\phi(\mathbf{u}, \mathbf{y})$ is nonlinear in $\mathbf{y}$.

The forthcoming analysis is conducted assuming that the cost estimates remain within the interval δ at steady-state operation, as illustrated in Figure 1. Based on a statistical description of v, δ could be selected by considering a desired confidence interval. Values that fall outside the selected confidence interval can simply be discarded.

Consider the kth iteration and the n_u past operating points, $\mathbf{u}_{k-j},\ j = 0, \ldots, n_u - 1$, and let us evaluate the cost as a function of the next operating point, which will generically be labeled $\mathbf{u}$. Using a first-order approximation of $\Phi_p(\mathbf{u}_{k-j})$ in the neighborhood of $\mathbf{u}$, the value of ψ at the past operating points is given by:

$$\begin{aligned}\psi(\mathbf{u}_{k-j}) = \Phi_p(\mathbf{u}_{k-j}) + v_{k-j} = (\psi(\mathbf{u}) - v) \\ + \frac{\partial \Phi_p}{\partial \mathbf{u}}(\mathbf{u})[\mathbf{u}_{k-j} - \mathbf{u}] + O\left(\|\mathbf{u}_{k-j} - \mathbf{u}\|^2\right) + v_{k-j}\end{aligned} \tag{5}$$

and, by neglecting the higher-order and noise terms:

$$\psi(\mathbf{u}_{k-j}) = \psi(\mathbf{u}) + \hat{\boldsymbol{\beta}}(\mathbf{u})[\mathbf{u}_{k-j} - \mathbf{u}], \tag{6}$$

where $\hat{\boldsymbol{\beta}}(\mathbf{u})$ is an estimate of the experimental cost gradient $\frac{\partial \Phi_p}{\partial \mathbf{u}}(\mathbf{u})$. $\hat{\boldsymbol{\beta}}(\mathbf{u})$ can be computed from the n_u past operating points $\mathbf{u}_k, \ldots,\ \mathbf{u}_{k-n_u+1}$ and the corresponding noisy cost values $\psi(\mathbf{u}_k), \ldots, \psi(\mathbf{u}_{k-n_u+1})$ by writing (6) in the following matrix form (Brdyś and Tatjewski [2005]):

$$\hat{\boldsymbol{\beta}}(\mathbf{u}) = \mathcal{Y}(\mathbf{u})\, \mathcal{U}^{-1}(\mathbf{u}) \tag{7}$$

with

$$\mathcal{U}(\mathbf{u}) := [\, \mathbf{u} - \mathbf{u}_k \ \ldots\ \mathbf{u} - \mathbf{u}_{k-n_u+1} \,] \in \mathbb{R}^{n_u \times n_u} \tag{8}$$

$$\mathcal{Y}(\mathbf{u}) := [\, \psi(\mathbf{u}) - \psi(\mathbf{u}_k) \ \ldots\ \psi(\mathbf{u}) - \psi(\mathbf{u}_{k-n_u+1}) \,] \tag{9}$$

The gradient error is defined as $\boldsymbol{\epsilon}(\mathbf{u}) := \hat{\boldsymbol{\beta}}(\mathbf{u}) - \frac{\partial \Phi_p}{\partial \mathbf{u}}(\mathbf{u})$, which, from (7) together with (4), can be split as $\boldsymbol{\epsilon}(\mathbf{u}) = \boldsymbol{\epsilon}^t(\mathbf{u}) + \boldsymbol{\epsilon}^n(\mathbf{u})$, with:

$$\begin{aligned}\boldsymbol{\epsilon}^t(\mathbf{u}) = [\, \Phi_p(\mathbf{u}) - \Phi_p(\mathbf{u}_k) \quad \ldots \\ \ldots \quad \Phi_p(\mathbf{u}) - \Phi_p(\mathbf{u}_{k-n_u+1}) \,]\, \mathcal{U}^{-1}(\mathbf{u}) - \frac{\partial \Phi_p}{\partial \mathbf{u}}(\mathbf{u})\end{aligned} \tag{10}$$

$$\boldsymbol{\epsilon}^n(\mathbf{u}) = [\, v - v_k \quad \ldots \quad v - v_{k-n_u+1} \,]\, \mathcal{U}^{-1}(\mathbf{u}) \tag{11}$$

where $\boldsymbol{\epsilon}^t$ and $\boldsymbol{\epsilon}^n$ represent the errors due to truncation and noise, respectively. Next, we investigate these two components of the gradient error.

Gradient Error due to Truncation. An upper bound on the norm of this error is given in the next proposition.

Proposition 1. Let $\Phi_p(\mathbf{u})$ be twice continuously differentiable with respect to $\mathbf{u}$. Then, given the n_u past operating points $\mathbf{u}_k, \ldots, \mathbf{u}_{k-n_u+1}$, an upper bound on $\|\boldsymbol{\epsilon}^t(\mathbf{u})\|$ is given by

$$\|\boldsymbol{\epsilon}^t(\mathbf{u})\| \leq \mathcal{E}^t(\mathbf{u}), \tag{12}$$

with

$$\mathcal{E}^t(\mathbf{u}) := \frac{\mathsf{d}_2}{2}\Big\| \big[\ (\mathbf{u}-\mathbf{u}_k)^\mathsf{T}(\mathbf{u}-\mathbf{u}_k)\ \ldots \tag{13}$$
$$\ldots\ (\mathbf{u}-\mathbf{u}_{k-n_u+1})^\mathsf{T}(\mathbf{u}-\mathbf{u}_{k-n_u+1})\ \big]\ \mathcal{U}^{-1}(\mathbf{u})\Big\|$$

where d_2 is the largest absolute eigenvalue of the Hessian of $\Phi_p(\cdot)$.

Proof. By Taylor series expansion of $\Phi_p(\mathbf{u}_{k-j})$ at $\mathbf{u}$ and upper bounding of the norm of the Hessian of Φ_p [Marchetti, 2009]. □

Note that d_2 represents an upper bound on the curvature of $\Phi_p(\cdot)$.

Gradient Error due to Measurement Noise. For relating the error norm $\|\boldsymbol{\epsilon}^n(\mathbf{u})\|$ to the location of the new operating point, the concepts of affine subspaces and distance between complement affine subspaces will be used (see Appendix A for a brief review of these concepts).

The largest possible value of $\|\boldsymbol{\epsilon}^n(\mathbf{u})\|$, noted $\|\boldsymbol{\epsilon}^n(\mathbf{u})\|_{\max}$, is computed in the next proposition.

Proposition 2. Given the n_u past operating points $\mathbf{u}_k, \ldots, \mathbf{u}_{k-n_u+1}$ and the interval δ for the noisy function $\psi(\cdot)$, the largest possible value of $\|\boldsymbol{\epsilon}^n(\mathbf{u})\|$ is

$$\|\boldsymbol{\epsilon}^n(\mathbf{u})\|_{\max} = \delta / l_{\min}(\mathbf{u}) \tag{14}$$

where $l_{\min}(\mathbf{u})$ is the shortest distance between all possible pairs of complement affine subspaces that can be generated from $\mathcal{S} = \{\mathbf{u}, \mathbf{u}_k, \ldots, \mathbf{u}_{k-n_u+1}\}$.

Proof. The proof proceeds in two parts: (i) the largest error occurs when the error v is either $\delta/2$ for some of the operating points and $-\delta/2$ for the other points, with each set of points defining an affine subspace; and (ii) the error vector $\boldsymbol{\epsilon}^n(\mathbf{u})$ is normal to both affine subspaces, which results in the largest possible error norm given by (14) [Marchetti, 2009]. □

5. UPPER BOUND ON GRADIENT ERROR

A bound on the condition number of the matrix $\mathcal{U}(\mathbf{u})$ was proposed in Brdyś and Tatjewski [1994, 2005]. This bound ensures that the new operating point does not introduce large errors in the gradient estimates due to ill-conditioning of $\mathcal{U}(\mathbf{u})$. However, the bound is not directly related to the errors resulting from truncation and measurement noise. This section introduces a consistent, although possibly conservative, upper bound on the gradient error norm.

Consider the desired upper bound $\mathcal{E}^U$ on the gradient error norm:

$$\|\boldsymbol{\epsilon}(\mathbf{u})\| \leq \|\boldsymbol{\epsilon}^t(\mathbf{u})\| + \|\boldsymbol{\epsilon}^n(\mathbf{u})\| \leq \mathcal{E}^U \tag{15}$$

Given the n_u past operating points $\mathbf{u}_k, \ldots, \mathbf{u}_{k-n_u+1}$, the following theorem provides a sufficient condition for the location of $\mathbf{u}$ so as to satisfy (15).

Theorem 1. (Sufficient condition for gradient accuracy). The gradient error norm $\|\boldsymbol{\epsilon}(\mathbf{u})\|$ does not exceed the desired upper bound $\mathcal{E}^U$ by choosing $\mathbf{u}$ that satisfies

$$\mathcal{E}(\mathbf{u}) := \mathcal{E}^t(\mathbf{u}) + \|\boldsymbol{\epsilon}^n(\mathbf{u})\|_{\max} \leq \mathcal{E}^U, \tag{16}$$

with $\mathcal{E}^t(\mathbf{u})$ and $\|\boldsymbol{\epsilon}^n(\mathbf{u})\|_{\max}$ given by (13) and (14), respectively.

Proof. The proof follows from (15), inequality (12) and the fact that $\|\boldsymbol{\epsilon}^n(\mathbf{u})\| \leq \delta / l_{\min}(\mathbf{u})$ from (14). □

For given values of δ and d_2, there is a minimal value that $\mathcal{E}(\mathbf{u})$ can take. Hence, $\mathcal{E}^U$ should be selected larger than this minimal value for the constraint (16) to be feasible.

Example 1. Consider the two-dimensional case ($n_u = 2$) with $\delta = 0.2$, $\mathsf{d}_2 = 2$ and the past operating points $\mathbf{u}_k = [0\ \ -0.5]^\mathsf{T}$ and $\mathbf{u}_{k-1} = [0\ \ 0.5]^\mathsf{T}$. The upper bounds $\mathcal{E}^t(\mathbf{u})$ and $\|\boldsymbol{\epsilon}^n(\mathbf{u})\|_{max}$ can be evaluated in terms of the location of the new operating point $\mathbf{u} = [u_1\ \ u_2]^\mathsf{T}$. Figures 2a-c show the contours of the error norms $\mathcal{E}^t(\mathbf{u})$, $\|\boldsymbol{\epsilon}^n(\mathbf{u})\|_{max}$ and $\mathcal{E}(\mathbf{u})$. It is seen that (i) both $\mathcal{E}^t(\mathbf{u})$ and $\|\boldsymbol{\epsilon}^n(\mathbf{u})\|_{max}$ increase as $\mathcal{U}(\mathbf{u})$ becomes more ill-conditioned ($\mathbf{u}$ aligned with $\mathbf{u}_k$ and $\mathbf{u}_{k-1}$), and (ii) the two regions generated by the constraint (16) *are nonconvex.*

Convex Constraint. The objective being to use the constraint (16) in the optimization problem (2), the fact that this constraint is nonconvex creates the possibility of multiple local solutions. Next, we introduce a tight relaxation that makes the constraint convex.

It can be seen that, for a given error level c, the expression $\mathcal{E}^t(\mathbf{u}) = c$ generates two $(n_u - 1)$-dimensional spheres of radius $r = \frac{c}{\mathsf{d}_2}$. The centers of these spheres are located symmetrically on each side of the hyperplane $\mathbf{n}_k^\mathsf{T}\mathbf{u} = b_k$ generated by the n_u past operating points $\mathbf{u}_k, \ldots, \mathbf{u}_{k-n_u+1}$. Considering the new operating point $\mathbf{u}$ located on the sphere, the center point is given by

$$\mathbf{u}_c^\mathsf{T}(\mathbf{u}) = \frac{1}{2}\Big[\ \mathbf{u}^\mathsf{T}\mathbf{u} - \mathbf{u}_k^\mathsf{T}\mathbf{u}_k\ \cdots \tag{17}$$
$$\mathbf{u}^\mathsf{T}\mathbf{u} - \mathbf{u}_{k-n_u+1}^\mathsf{T}\mathbf{u}_{k-n_u+1}\ \Big]\ \mathcal{U}^{-1}(\mathbf{u}).$$

It can be shown that $\|\boldsymbol{\epsilon}^n(\mathbf{u})\|_{\max}$ is convex on each side of the hyperplane $\mathbf{n}_k^\mathsf{T}\mathbf{u} = b_k$ (see also Figure 2b). Hence, non-convexity of the constraint (16) is due to the part of the aforementioned spheres that crosses the hyperplane $\mathbf{n}_k^\mathsf{T}\mathbf{u} = b_k$. The distance (positive or negative) from the center point $\mathbf{u}_c(\mathbf{u})$ to the hyperplane $\mathbf{n}_k^\mathsf{T}\mathbf{u} = b_k$ is given by:

$$l_C(\mathbf{u}) = \frac{b_k - \mathbf{n}_k^\mathsf{T}\mathbf{u}_c(\mathbf{u})}{\|\mathbf{n}_k\|}. \tag{18}$$

Given the n_u operating points $\mathbf{u}_k, \ldots, \mathbf{u}_{k-n_u+1}$, the point $\mathbf{u}_{\mathrm{m},k}$ can be obtained by projecting the center point $\mathbf{u}_c(\mathbf{u})$ on the hyperplane $\mathbf{n}_k^\mathsf{T}\mathbf{u} = b_k$:

$$\mathbf{u}_{\mathrm{m},k} = \mathbf{u}_c(\mathbf{u}) + \frac{l_C(\mathbf{u})}{\|\mathbf{n}_k\|}\mathbf{n}_k \tag{19}$$

It appears that $\mathbf{u}_{\mathrm{m},k}$ is independent of $\mathbf{u}$. For a given upper bound $\mathcal{E}^U$, it is then possible to define convex feasible

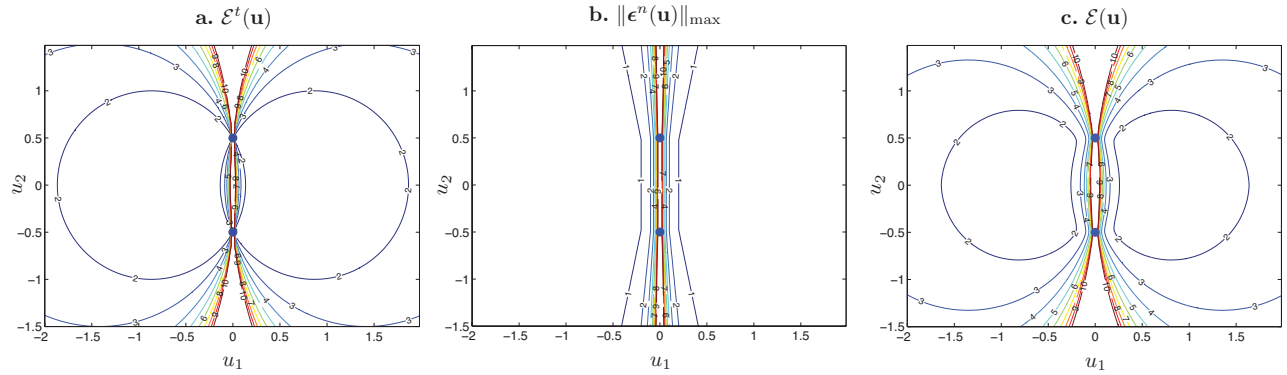


Fig. 2. Contour maps for the norm of the gradient error due to (**a**) truncation error, (**b**) measurement noise, and (**c**) total error.

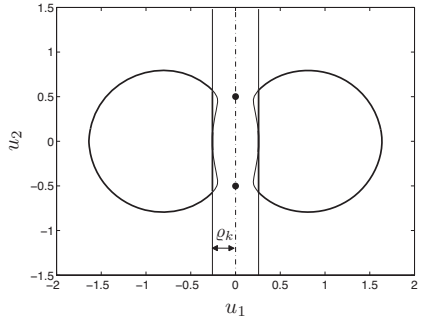


Fig. 3. Convex regions (in bold) corresponding to the constraint $\mathcal{E}(\mathbf{u}) \leq 2$.

regions by adding constraints expressing the minimal distance between the new operating point $\mathbf{u}$ and the hyperplane, which eliminates the non-convex part of the regions generated by (16), as illustrated in Figure 3. The minimal point-to-hyperplane distance ϱ_k can be determined numerically by finding the smallest absolute value solution of the following equation:

$$\mathcal{E}\left(\mathbf{u}_{\mathrm{m},k} + \frac{\varrho_k}{\|\mathbf{n}_k\|}\mathbf{n}_k\right) = \mathcal{E}^U$$

6. DUAL MODIFIER-ADAPTATION SCHEME

The dual modifier-adaptation scheme proposed in this section uses the upper bound on the gradient error defined in Section 5 as a constraint in the optimization problem (2). On each side of the hyperplane $\mathbf{n}_k^\mathsf{T}\mathbf{u} = b_k$ generated by the n_u past operating points, a modified model-based optimization problem is solved. The optimization problem corresponding to the half space $\mathbf{n}_k^\mathsf{T}\mathbf{u} > b_k$ reads:

$$\mathbf{u}_{k+1}^+ = \arg\min_{\mathbf{u}} \Phi_m(\mathbf{u},\boldsymbol{\theta}) = \Phi(\mathbf{u},\boldsymbol{\theta}) + \boldsymbol{\lambda}_k^\mathsf{T}\mathbf{u} \qquad (20)$$
$$\text{s.t.} \quad \mathcal{E}(\mathbf{u}) \leq \mathcal{E}^U, \qquad \mathbf{n}_k^\mathsf{T}\mathbf{u} > b_k + \varrho_k\|\mathbf{n}_k\|$$

while, for the half space $\mathbf{n}_k^\mathsf{T}\mathbf{u} < b_k$, one has:

$$\mathbf{u}_{k+1}^- = \arg\min_{\mathbf{u}} \Phi_m(\mathbf{u},\boldsymbol{\theta}) = \Phi(\mathbf{u},\boldsymbol{\theta}) + \boldsymbol{\lambda}_k^\mathsf{T}\mathbf{u} \qquad (21)$$
$$\text{s.t.} \quad \mathcal{E}(\mathbf{u}) \leq \mathcal{E}^U, \qquad \mathbf{n}_k^\mathsf{T}\mathbf{u} < b_k - \varrho_k\|\mathbf{n}_k\|$$

The modifiers $\boldsymbol{\lambda}_k^\mathsf{T}$ are adapted as in (3). The next operating point is chosen as the value of $\{\mathbf{u}_{k+1}^+, \mathbf{u}_{k+1}^-\}$ that minimizes the augmented cost function $\Phi_m(\mathbf{u},\boldsymbol{\theta})$.

7. OPTIMIZATION OF REACTOR OPERATION

The reactor in the Williams-Otto plant (Williams and Otto [1960]), as modified by Roberts [1979], is used to illustrate the dual modifier-adaptation scheme. This reactor example has also been used to illustrate model adequacy and RTO performance (Forbes et al. [1994], Zhang and Forbes [2000]). It consists of an ideal CSTR in which the following reactions occur:

$$\begin{aligned} A + B &\longrightarrow C & \bar{k}_1 &= 1.660 \times 10^6 e^{-6666.7/(T_R+273.15)} \\ C + B &\longrightarrow P + E & \bar{k}_2 &= 7.212 \times 10^8 e^{-8333.3/(T_R+273.15)} \\ C + P &\longrightarrow G & \bar{k}_3 &= 2.675 \times 10^{12} e^{-11111/(T_R+273.15)} \end{aligned}$$

where the reactants A and B are fed with the mass flowrates F_A and F_B, respectively. The desired products are P and E. C is an intermediate product and G is an undesired product. The product stream has the mass flowrate $F = F_A + F_B$. Operation is isothermal at the temperature T_R. The reactor mass holdup is 2105 kg.

The objective is to maximize profit, which is expressed as the cost difference between the products and the reactants:

$$\phi(\mathbf{u},\mathbf{y}) = 1143.38X_P F + 25.92X_E F - 76.23F_A - 114.34F_B$$

The flowrate of reactant A is fixed at 1.8275 kg/s. The flowrate of reactant B and the reactor temperature are the decision variables, thus $\mathbf{u} = [F_B \; T_R]^\mathsf{T}$.

In this example, the aforementioned reaction scheme corresponds to the simulated reality. However, since it is assumed that the reaction scheme is not well understood, the following two reactions have been proposed to model the system (Forbes et al. [1994]):

$$\begin{aligned} A + 2B &\longrightarrow P + E & k_1 &= 2.189 \times 10^8 e^{-8077.6/(T_R+273.15)} \\ A + B + P &\longrightarrow G & k_2 &= 4.310 \times 10^{13} e^{-12438/(T_R+273.15)} \end{aligned}$$

The material balance equations for the plant and the approximate model can be found in Zhang and Forbes [2000].

The inputs are scaled using the intervals $[3, 6]$ for F_B, and $[70, 100]$ for T_R. In this range, the maximal value of d_2 obtained with the scaled inputs is $\mathsf{d}_2 = 1030$ for the model, whereas the (unknown) plant value is $\mathsf{d}_2 = 1221$. The simulations are carried out assuming that the noise v has a Gaussian distribution with standard deviation $\sigma_\phi = 0.5$. The noise interval $\delta = 3$ is chosen. The exponential filter (3) is implemented with $d = 0.5$.

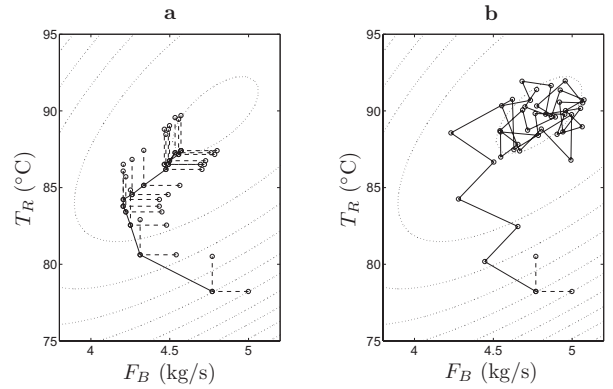


Fig. 4. Input trajectories for 45 operating points. The dotted lines represent the contours of the plant cost function. (**a**) Modifier adaptation using FFD. (**b**) Dual modifier adaptation with bound on gradient error.

Modifier Adaptation using FFD. First, modifier adaptation is applied using the FFD approach, which consists in perturbing the inputs one at the time from the current operating point with the fixed step size h. The gradient error norm, which is a function of h, is found to be minimal for $h^\star = 0.0763$ (scaled value). The corresponding gradient error norm $\mathcal{E}^t(h^\star) + \|\boldsymbol{\epsilon}^n(h^\star)\|_{\max}$ is 111.2 (Marchetti [2009]). Figure 4a shows a resulting input trajectory. The observed offset with respect to the plant optimum results mainly from the gradient error due to truncation.

Dual Modifier Adaptation with Bound on Gradient Error. Dual modifier adaptation is applied with $\mathcal{E}^U = 111.2$ (same value as above). The algorithm is initialized using FFD. Figure 4b shows a resulting input trajectory. Compared with modifier adaptation using FFD, significantly fewer operating points are required to approach the optimum.

Figure 5a shows the evolution of the plant profit and the gradient error norm for 20 noise realizations. At iteration 20, the flowrate F_A is increased from 1.8275 kg/s to 2.2 kg/s. Modifier adaptation tracks the change in the plant optimum. It can be seen in the upper plot of Figure 5a that the neighborhood of the new optimal profit is reached within 6 iterations for all 20 realizations. Also, the lower plot of Figure 5a shows that the gradient error norm is kept below $\mathcal{E}^U$. The observed peak in gradient error occurring at iterations 21 and 22 is due to the fact that, at these points, the gradient is inconsistent in that it is estimated using operating points with different values of F_A.

Dual Modifier Adaptation with Bound on Condition Number. For the sake of comparison, dual modifier adaptation is also applied with a lower bound on the inverse condition number of $\mathcal{U}(\mathbf{u})$, as proposed in Brdyś and Tatjewski [1994, 2005]. The results are shown in Figure 5b. A lower bound of 0.4 gives an adaptation that is similar to that using the gradient error bound in the first iterations. However, as soon as the neighborhood of the plant optimum is reached, the distance between the operating points decreases, and the gradient estimates become much less accurate. Furthermore, the feasible regions given by the condition number constraint decrease proportionally

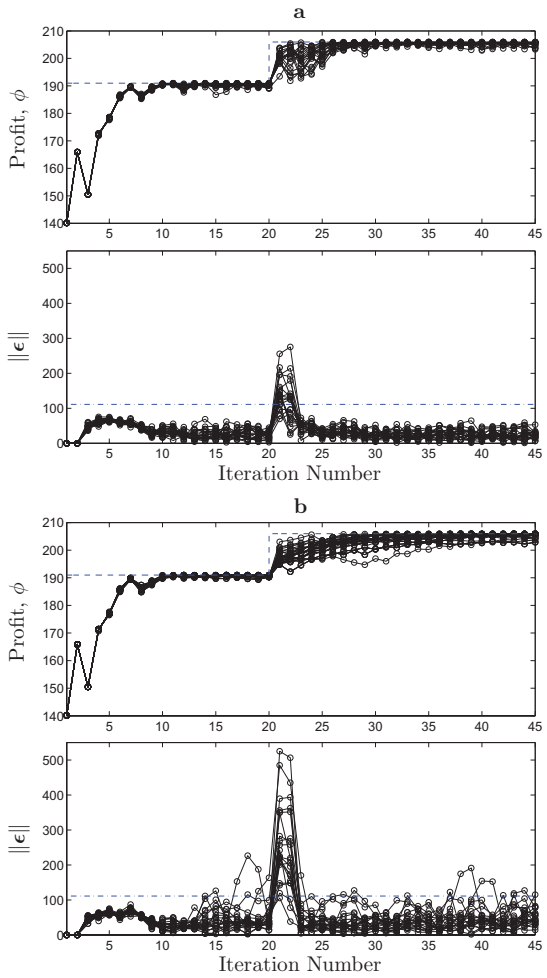


Fig. 5. Optimization for 20 noise realizations; there is a flowrate change at iteration 20. (**a**) Dual modifier adaptation with bound on gradient error norm. (**b**) Dual modifier adaptation with bound on the condition number. Dashed line: Optimal profit for the plant. Dash-dotted line: $\mathcal{E}^U = 111.2$.

to the distance between points. This appropriately prevents taking large steps in the wrong direction, but it also appears less suitable for tracking a changing optimum.

8. CONCLUSIONS

This study has demonstrated the potential of dual modifier adaptation, which pays attention to the accuracy with which the gradients are estimated. The results of the case study indicate that this approach, wherein the gradient error norm is bounded, produces more accurate gradient estimates than with simply bounding the condition number of $\mathcal{U}(\mathbf{u})$, i.e. a measure of the relative position of the successive inputs. In addition, the proposed scheme seems more capable of tracking a changing optimum. The performance depends on the amount of plant-model mismatch, the noise level, the estimated curvature of the cost function d_2, and the filter parameter d.

Future work will consider the extension of this approach to constrained optimization problems. In this case, mod-

ifier adaptation will require an estimate of the cost and constraint gradients of the plant to be available at each iteration. In order to be able to use the upper bound on the gradient error developed in this paper, a possible way is to associate the parameters δ and d_2 to a Lagrangian function, which represents a linear combination of the cost and constraint functions.

REFERENCES

Brdyś, M. and Tatjewski, P. (1994). An algorithm for steady-state optimizing dual control of uncertain plants. In *Proc. 1st IFAC Workshop on New Trends in Design of Control Systems*, 249–254. Smolenice, Slovakia.

Brdyś, M. and Tatjewski, P. (2005). *Iterative Algorithms for Multilayer Optimizing Control*. Imperial College Press, London UK.

Chachuat, B., Srinivasan, B., and Bonvin, D. (2009). Adaptation strategies for real-time optimization. *Comp. Chem. Eng. (in press)*.

Forbes, J.F., Marlin, T.E., and MacGregor, J.F. (1994). Model adequacy requirements for optimizing plant operations. *Comp. Chem. Eng.*, 18(6), 497–510.

Gao, W. and Engell, S. (2005). Iterative set-point optimization of batch chromatography. *Comp. Chem. Eng.*, 29, 1401–1409.

Mansour, M. and Ellis, J.E. (2003). Comparison of methods for estimating real process derivatives in online optimization. *App. Math. Modelling*, 27, 275–291.

Marchetti, A. (2009). *Modifier-Adaptation Methodology for Real-Time Optimization*. Ph.D. thesis, École Polytechnique Fédérale de Lausanne (in preparation).

Marchetti, A., Chachuat, B., and Bonvin, D. (2009). Modifier adaptation methodology for real-time optimization. *Ind. Eng. Chem. Res.*, DOI: 10.1021/ie801352x.

Marlin, T.E. and Hrymak, A.N. (1997). Real-time operations optimization of continuous processes. In *AIChE Symposium Series - CPC-V*, volume 93, 156–164.

Roberts, P.D. (1979). An algorithm for steady-state system optimization and parameter estimation. *Int. J. Systems Sci.*, 10, 719–734.

Williams, T.J. and Otto, R.E. (1960). A generalized chemical processing model for the investigation of computer control. *AIEE Trans.*, 79, 458.

Zhang, Y. and Forbes, J.F. (2000). Extended design cost: A performance criterion for real-time optimization systems. *Comp. Chem. Eng.*, 24, 1829–1841.

Appendix A. AFFINE SUBSPACES

In a n_u-dimensional space, a point is an affine subspace of dimension 0, a line is an affine subspace of dimension 1, and a plane is an affine subspace of dimension 2. An affine subspace of dimension $(n_u - 1)$ is called an hyperplane.

Hyperplane. An hyperplane in n_u-dimensional space is given by

$$n_1 u_1 + n_2 u_2 + \cdots + n_{n_u} u_{n_u} = b, \quad \text{or:} \quad \mathbf{n}^\mathsf{T}\mathbf{u} = b \quad \text{(A.1)}$$

and divides the space into two half-spaces: $\mathbf{n}^\mathsf{T}\mathbf{u} > b$, and $\mathbf{n}^\mathsf{T}\mathbf{u} < b$.

Complement affine subspaces. Given a set of $(n_u + 1)$ points in a n_u-dimensional space, $\mathcal{S} := \{\mathbf{u}_1, \ldots, \mathbf{u}_{n_u+1}\}$, a proper subset $\mathcal{S}^A$, i.e. $\mathcal{S}^A \subsetneq \mathcal{S}$, of $n_u^A \in \{1, \ldots, n_u\}$ points generates an affine subspace of dimension $(n_u^A - 1)$:

$$\mathbf{u} = \mathbf{u}_1 + \lambda_{1,2}\frac{\mathbf{u}_1 - \mathbf{u}_2}{\|\mathbf{u}_1 - \mathbf{u}_2\|} + \cdots + \lambda_{1,n_u^A}\frac{\mathbf{u}_1 - \mathbf{u}_{n_u^A}}{\|\mathbf{u}_1 - \mathbf{u}_{n_u^A}\|} \quad \text{(A.2)}$$

where the parameters $\lambda_{1,2}, \ldots, \lambda_{1,n_u^A}$ represent distances from the point $\mathbf{u}_1$ in the directions $\mathbf{u}_1 - \mathbf{u}_2, \ldots, \mathbf{u}_1 - \mathbf{u}_{n_u^A}$, respectively. The complement subset $\mathcal{S}^C := \mathcal{S} \setminus \mathcal{S}^A$ of $(n_u + 1 - n_u^A)$ points, generates the complement affine subspace of dimension $(n_u - n_u^A)$:

$$\mathbf{u} = \mathbf{u}_{n_u^A+1} + \lambda_{n_u^A+1,n_u^A+2}\frac{\mathbf{u}_{n_u^A+1} - \mathbf{u}_{n_u^A+2}}{\|\mathbf{u}_{n_u^A+1} - \mathbf{u}_{n_u^A+2}\|} + \ldots \quad \text{(A.3)}$$
$$\cdots + \lambda_{n_u^A+1,n+1}\frac{\mathbf{u}_{n_u^A+1} - \mathbf{u}_{n_u+1}}{\|\mathbf{u}_{n_u^A+1} - \mathbf{u}_{n_u+1}\|}$$

Distance between complement affine subspaces.

Definition 1. (Distance between complement affine subspaces). Given a set of (n_u+1) points in a n_u-dimensional space, $\mathcal{S} := \{\mathbf{u}_1, \ldots, \mathbf{u}_{n_u+1}\}$, a proper subset of $\mathcal{S}$, $\mathcal{S}^A \subsetneq \mathcal{S}$ of $n_u^A \in \{1, \ldots, n_u\}$ points, and its complement $\mathcal{S}^C := \mathcal{S} \setminus \mathcal{S}^A$ of $(n_u + 1 - n_u^A)$ points, the distance between complement affine subspaces is defined as the (orthogonal) distance between the affine subspace of dimension $(n_u^A - 1)$ generated by all the points in $\mathcal{S}^A$, and the affine subspace of dimension $(n_u - n_u^A)$ generated by all the points in $\mathcal{S}^C$.

The total number of possible pairs of complement affine subspaces that can be generated from $\mathcal{S}$ is $n_b = 1 + \sum_{s=1}^{n_u-1} 2^s$.

Definition 2. (Nearest complement affine subspaces). The shortest distance between complement affine subspaces is given by $l_{\min} := \min\{l_1, l_2, \ldots, l_{n_b}\}$, where $l_1, l_2, \ldots, l_{n_b}$ are the distances between all possible pairs of complement affine subspaces that can be generated from $\mathcal{S}$.

In the 2-dimensional case ($n_u = 2$), the number of distances to evaluate is $n_b = 3$, which corresponds to the 3 point-to-line distances. In the 3-dimensional case, there are $n_b = 7$ distances to evaluate, which correspond to 4 point-to-plane distances, and 3 line-to-line distances.

In order to compute the distance between the complement affine subspaces (A.2) and (A.3), a vector $\mathbf{n}$ that is normal to both subspaces is required:

$$[\ \mathbf{u}_1 - \mathbf{u}_2\ \ldots\ \mathbf{u}_1 - \mathbf{u}_{n_u^A}\ \ \mathbf{u}_{n_u^A+1} - \mathbf{u}_{n_u^A+2}\ \cdots \quad \text{(A.4)}$$
$$\mathbf{u}_{n_u^A+1} - \mathbf{u}_{n_u+1}\ ]^\mathsf{T}\mathbf{n} = \mathbf{0}, \qquad \text{or,} \qquad \mathsf{U}\mathbf{n} = \mathbf{0}.$$

The matrix $\mathsf{U} \in \mathbb{R}^{(n_u-1)\times n_u}$ is of rank $(n_u - 1)$. The vector $\mathbf{n}$ can be obtained by singular-value decomposition of U.

Given a point $\mathbf{u}^a$ that belongs to the affine subspace (A.2), a point $\mathbf{u}^b$ that belongs to the complement affine subspace (A.3), and a vector $\mathbf{n}$ that is normal to both complement affine subspaces, the distance l_{AC} between the two complement affine subspaces is:

$$l_{AC} = \frac{|\mathbf{n}^\mathsf{T}(\mathbf{u}^b - \mathbf{u}^a)|}{\|\mathbf{n}\|} \quad \text{(A.5)}$$

Optimally Invariant Variable Combinations for Nonlinear Systems

Johannes E. P. Jäschke, Sigurd Skogestad

Norwegian University of Science and Technology (NTNU), Trondheim, Norway (e-mail: skoge@chemeng.ntnu.no).

Abstract: In this article we present an "explicit RTO" approach for achieving optimal steady state operation without requiring expensive online calculations. After identifying regions of constant active constraints, it is shown that there exist some optimally invariant variable combination for each region. If the unknown variables can be eliminated by measurements and system equations, the invariant combinations can be used for control. Moreover, we show that the measurement invariants can be used for detecting changes in the active set and for finding the right region to switch to. This explicit RTO approach is applied to a CSTR described by a set of rational equations. We show how the invariant variable combinations are derived, and use polynomial reduction to eliminate the unknown variables to obtain the measurement invariants which are used for control.

Keywords: Optimizing control, Polynomial systems, Real-time optimization, Explicit RTO, Self-optimizing control, Optimally invariant measurement combinations, Changing active sets

1. INTRODUCTION

Optimal operation of chemical processes becomes increasingly important in order to be able to compete in the international markets and to minimize environmental impact. A well established tool to achieve this goal is real-time optimization (RTO), where the optimal set-points are computed on-line, based on measurements taken at given sample times. This involves setting up and maintaining a real-time computation system, which can be very expensive and time consuming.

An alternative approach is to use off-line calculations and analysis to minimize or avoid complex on-line computations for example by finding optimally invariant measurement combinations ('self-optimizing' variable combinations, (Narasimhan and Skogestad (2007)). Controlling these combinations to their setpoints guarantees to operate the process optimal or close to optimal, with a certain acceptable loss (Skogestad (2000)). The combinations can be controlled by a simple control structure based on PI controllers. The conventional real-time optimization problem can either be replaced completely or partially by controlling invariant variable combinations. In practice, many processes are operated by something similar to this alternative approach, although not always consciously. That is, the optimization problem is not formulated explicitly and the control variables are chosen from experience and engineering intuition.

This publication presents two main results. The first one is extending the idea of self-optimizing control from unconstrained linear problems to constrained nonlinear problems. To the authors knowledge, optimally invariant variable combinations have been considered systematically only for linear plants with quadratic performance index (see e.g. Alstad et al. (2009)). A second contribution is the proof that using controlled variable to identify new sets of active constraints will always identify the correct active set. Although measurement invariants have been used before for active set identification (Manum et al. (2007)), it has not been proved that this holds for nonlinear problems, too.

2. GENERAL PROCEDURE

We consider a plant at steady state and assume the plant performance can be modelled as an optimization problem with a performance index J together with equality and inequality constraints, $g(\mathbf{u}, \mathbf{x}, \mathbf{d})$ and $h(\mathbf{u}, \mathbf{x}, \mathbf{d})$:

$$\min_{\mathbf{u},\mathbf{x}} J \quad \text{s.t} \quad \begin{cases} g(\mathbf{u}, \mathbf{x}, \mathbf{d}) = 0 \\ h(\mathbf{u}, \mathbf{x}, \mathbf{d}) \leq 0 \end{cases} \tag{1}$$

The variables $\mathbf{u}$, $\mathbf{x}$, $\mathbf{d}$ denote the manipulated input variables, the internal states, and the disturbance variables, respectively. In addition, we assume that there are measurements $\mathbf{y}(\mathbf{x}, \mathbf{u}, \mathbf{d})$, which provide information about the internal states and the disturbances of the process.

In order to obtain optimal operation we do not optimize the model on-line at given sample times. Instead, we use the structure of the problem to find optimally invariant variable combinations for the system. Since the available number of degrees of freedom changes when an inequality constraint becomes active, we have to find a new set of invariant measurement combinations for each set of constraints that becomes active during operation of the plant. This makes it necessary to define separate control structures for each region. Therefore, the first step is to partition the operating space into regions defined by the set of active constraints, i.e. the system is optimized for all possible disturbances $\mathbf{d}$ and the active constraints in each region are identified.

In the second step, we determine (nonlinear) variable combinations which yield optimal operation when kept at their constant setpoint. The variables resulting from this step cannot be used for control directly, because they contain unknown disturbance variables and internal states which are not known. To be able to control the system, we attempt to "model" the variable invariants by expressions which only contain known variables. These can then be used for control in feedback loops.

The last step in this procedure is to define rules for detecting and switching regions when the active constraints change. In many cases this can be done by monitoring the controlled variables of the neighbouring region and switching when the controlled variable of the neighbouring region reaches its optimal value.

3. DETERMINING INVARIANT VARIABLE COMBINATIONS

3.1 Invariants for systems with quadratic objective and linear inequality constraints and linear measurements

To illustrate the idea of finding invariant variable combinations we first consider a problem with a quadratic objective and linear constraints. After having identified n_r regions of active constraints, we can define an equality constrained optimization problem for each region.

Given $\mathbf{z} \in \mathbb{R}^{n_z \times 1}$ and $\mathbf{d} \in \mathbb{R}^{n_d \times 1}$, consider the constrained optimization problem:

$$\min_{\mathbf{z}} J = \min_{\mathbf{z}} [\mathbf{z^T}\ \mathbf{d^T}] \begin{bmatrix} \mathbf{J_{zz}} & \mathbf{J_{zd}} \\ \mathbf{J_{zd}}^{\mathbf{T}} & \mathbf{J_{dd}} \end{bmatrix} \begin{bmatrix} \mathbf{z} \\ \mathbf{d} \end{bmatrix} \tag{2}$$

subject to

$$\mathbf{A}_z \mathbf{z} + \mathbf{A}_d \mathbf{d} = \tilde{\mathbf{A}} \begin{bmatrix} \mathbf{z} \\ \mathbf{d} \end{bmatrix} = 0 \tag{3}$$

where we have $\mathbf{A}_z \in \mathbb{R}^{n_c \times n_z}$ has rank n_c, $\mathbf{A}_d \in \mathbb{R}^{n_c \times n_d}$, $\tilde{\mathbf{A}} = [\mathbf{A}_z\ \mathbf{A}_d]$, and $\mathbf{J_{zz}} > 0$.

Eq. (3) may include the model equations as well as active (equality) constraints. Instead of using (3) to eliminate n_c internal states to obtain an unconstrained problem, we keep the constraints explicit in the formulation as this more general formulation will be used later when presenting the nonlinear case (where the internal states are not easily substituted). The Karush-Kuhn-Tucker first order optimality conditions give

$$\nabla_z L = \nabla_z J + \mathbf{A}_z^{\mathbf{T}} \lambda = \tilde{J} \begin{bmatrix} \mathbf{z} \\ \mathbf{d} \end{bmatrix} + \mathbf{A}_z^{\mathbf{T}} \lambda = 0, \tag{4}$$

where $\tilde{J} = [\mathbf{J_{uu}}\ \mathbf{J_{ud}}]$, and λ is the vector of Lagrangian multipliers. Therefore, from (4) we have that

$$\mathbf{A}_z^{\mathbf{T}} \lambda = -\tilde{J} \begin{bmatrix} \mathbf{z} \\ \mathbf{d} \end{bmatrix}. \tag{5}$$

$\mathbf{A}_z$ is not full column rank, so let $\mathbf{N}_z$ be a basis for the null space of $\mathbf{A}_z$ with dimension $n_{DOF} = n_z - n_c$. Then $\mathbf{N}_z^{\mathbf{T}} \mathbf{A}_z^{\mathbf{T}} = 0$, and at the optimum we must have

$$\mathbf{c}^v(\mathbf{z}, \mathbf{d}) \triangleq \mathbf{N}_z^{\mathbf{T}} \tilde{J} \begin{bmatrix} \mathbf{z} \\ \mathbf{d} \end{bmatrix} = 0 \tag{6}$$

for the system (5) to be uniquely solvable for λ. Keeping $\mathbf{c}^v(\mathbf{z}, \mathbf{d})$ at zero (in addition to the active constraints), is always optimal. However, it cannot be used for control directly, as it contains unknown (unmeasured) variables. For control, we need a function of measurements $\mathbf{c}(\mathbf{y})$, such that the difference between the invariant and the measurement combination is minimal. Here, we want to "model" $\mathbf{c}^v(\mathbf{z}, \mathbf{d})$ perfectly, such that

$$\mathbf{c}(\mathbf{y}) = \mathbf{c}^v(\mathbf{z}, \mathbf{d}). \tag{7}$$

Then controlling $\mathbf{c}(\mathbf{y}) = 0$ yields optimal operation. If we have $n_z + n_d$ independent linear measurements

$$y = \mathbf{G}^y \begin{bmatrix} \mathbf{z} \\ \mathbf{d} \end{bmatrix}, \tag{8}$$

where $\mathbf{G}^y$ is invertible, we can use them with (6) to give

$$\mathbf{c}(\mathbf{y}) = \mathbf{N}^{\mathbf{T}} \tilde{J} [\mathbf{G}^y]^{-1} \mathbf{y}. \tag{9}$$

However, note that we actually only need $n_z - n_c + n_d = n_{DOF} + n_d$ measurements, since the model equations (3) can be used to eliminate the constrained degrees of freedom (internal states). This is shown in Appendix A.

Remark 1. In the unconstrained case, the optimal invariant variable combination is simply the gradient, such that we have $\mathbf{c}(\mathbf{y}) = \mathbf{H}\mathbf{y} = \nabla_u J$, and $\mathbf{H} = \tilde{J}[\tilde{\mathbf{G}}^y]^{-1}$.

3.2 Invariants for polynomial and rational systems

An analog approach may be taken for obtaining invariant variable combinations for more general systems described by polynomials. Since rational equations can be transformed into polynomials by multiplying with the common denominator, the method is applicable to rational systems, too.

Initially, all regions defined by constant active constraints are determined. For each region we then have:

Theorem 1. (Nonlinear invariants). Given $\mathbf{z}, \mathbf{d}$ as in section 3.1, consider the nonlinear optimization problem

$$\min_{\mathbf{z}} J(\mathbf{z}, \mathbf{d}) \quad \text{s.t} \quad g_i(\mathbf{z}, \mathbf{d}) = 0, \quad i = 1 \dots n_g, \tag{10}$$

and implicit measurement relations

$$m_j(\mathbf{y}, \mathbf{z}, \mathbf{d}) = 0 \quad j = 1 \dots n_y, \tag{11}$$

where $\mathbf{y}$ is the measured variable. If the Jacobian $\mathbf{A}_z(\mathbf{z}, \mathbf{d}) = [\nabla_z g]$ has full rank n_g at the optimum throughout the region, following holds:

(1) There exist $n_{DOF} = n_z - n_g$ independent invariant variable combinations $\mathbf{c}^v$ with

$$\mathbf{c}^v = [\mathbf{N}_z(\mathbf{z}, \mathbf{d})]^{\mathbf{T}} \nabla_z J(\mathbf{z}, \mathbf{d}), \tag{12}$$

where $\mathbf{N}_z(\mathbf{z}, \mathbf{d})$ denotes the null space of the Jacobian of the active constraints $g(\mathbf{z}, \mathbf{d})$.

(2) If there exist polynomials $\alpha_i(\mathbf{z}, \mathbf{d})$ and $\beta_j(\mathbf{z}, \mathbf{d})$, such that element of $\mathbf{c}^v$ can be expressed by

$$c^v = \sum_{i,j} (\alpha_i(\mathbf{z}, \mathbf{d}) g_i(\mathbf{z}, \mathbf{d}) + \beta_j(\mathbf{z}, \mathbf{d}) m_j(\mathbf{y}, \mathbf{z}, \mathbf{d})) + c(\mathbf{y}), \tag{13}$$

then the term $c(\mathbf{y})$ is the desired self-optimizing variable which when controlled to zero yields optimal operation.

Proof. Calculate the Jacobian of the constraints:

$$\mathbf{A}_z(\mathbf{z}, \mathbf{d}) = \left[\ [\nabla_z g_1(\mathbf{z}, \mathbf{d})]^{\mathbf{T}}, \cdots, [\nabla_z g_{n_g}(\mathbf{z}, \mathbf{d})]^{\mathbf{T}}\ \right]^{\mathbf{T}} \tag{14}$$

Since $\mathbf{A}_z(\mathbf{z},\mathbf{d})$ has full rank, the null space has a constant dimension and there exist a unique vector λ which satisfies the KKT conditions (Nocedal and Wright (2006)):

$$\begin{aligned} \nabla_z J(\mathbf{z},\mathbf{d}) + [\mathbf{A}_z(\mathbf{z},\mathbf{d})]^{\mathbf{T}} \lambda = 0 \\ g_i(\mathbf{z},\mathbf{d}) = 0, \quad i = 1 \dots n_g \end{aligned} \tag{15}$$

For the existence and uniqueness of λ, we always must have, that

$$[\mathbf{N}_z(\mathbf{z},\mathbf{d})]^{\mathbf{T}} \nabla_z J = -[\mathbf{N}_z(\mathbf{z},\mathbf{d})]^{\mathbf{T}} [\mathbf{A}(\mathbf{z},\mathbf{d})]^{\mathbf{T}} \lambda, \tag{16}$$

where $\mathbf{N}_z(\mathbf{z},\mathbf{d})$ is chosen as a basis for the $n_z - n_g$-dimensional null space of $\mathbf{A}_z(\mathbf{z},\mathbf{d})$. The optimal invariant variable combination to be kept at $\mathbf{c}^v = 0$ is then given by:

$$\mathbf{c}^v = [\mathbf{N}_z(\mathbf{z},\mathbf{d})]^{\mathbf{T}} \nabla_z J(\mathbf{z},\mathbf{d}) \tag{17}$$

The second statement follows from the implicit relations $g_i(\mathbf{z},\mathbf{d}) = 0$ and $m_j(\mathbf{y},\mathbf{z},\mathbf{d}) = 0$. □

Remark 2. This variable combination (17) is not unique in the sense that it can be premultiplied by any nonsingular matrix while still yielding optimal operation. However, since the variable combination is derived from the KKT conditions, it is unique in the sense that for a convex optimization problem $\mathbf{c}^v = 0$ if the system is operated optimally, because the stationary point is unique.

Remark 3. The full rank assumption for $\mathbf{A}_z(\mathbf{z},\mathbf{d})$ at the optimum is valid in most practical cases, as rank deficiency at the optimum implies that the degrees of freedom in the problem change.

Remark 4. The functions α and β can be found using Gröbner theory (Cox et al. (1992)), and the condition for the existence of polynomial functions α_i and β_i is that every variable to be eliminated must appear in the initial ideal generated by all g_i and m_j in lower or equal degree than in $\mathbf{c}^v$.

To illustrate the concept, we present a toy example.

Example 1. (Nonlinear invariants). We consider a cost function $J(z_1, z_2, d) = z_1^2 + z_2^2$ subject to the constraint

$$g = (z_1 - 1)^2 + (z_2 - d)^2 - 5. \tag{18}$$

In addition to the known z_1, z_2, the system has one measurement y with the measurement relation

$$m = z_1 z_2 + z_1 d - y = 0. \tag{19}$$

First we calculate the Jacobian of (18) with respect to z

$$\mathbf{A}(z_1, z_2, d) = [\, 2(z_1 - 1) \;\; 2(z_2 - d) \,], \tag{20}$$

and the basis of its null space:

$$\mathbf{N}(z_1, z_2, d) = [\, -(z_2 - d) \;\; (z_1 - 1) \,]^{\mathbf{T}}. \tag{21}$$

After computing the gradient of the cost function J

$$\nabla J = [\, 2z_1 \;\; 2z_2 \,]^{\mathbf{T}}, \tag{22}$$

we obtain the invariant variable combination as in (17):

$$c^v = \underbrace{[\, -(z_2 - d) \;\; (z_1 - 1) \,]}_{[\mathbf{N}(\mathbf{z},\mathbf{d})]^{\mathbf{T}}} \underbrace{\begin{bmatrix} 2z_1 \\ 2z_2 \end{bmatrix}}_{\nabla J} = 2(z_1 d - z_2) \tag{23}$$

However, c^v contains the unmeasured disturbance d, so it cannot be used for control. Using the measurement relation (19) and equation (13) we see by inspection that $\alpha = 0$ and $\beta = 2$ yield a $c(y)$ which satisfies (7):

$$\begin{aligned} c^v = \underbrace{2(z_1 d - z_2)}_{c^v} = \underbrace{0}_{\alpha} g + \underbrace{2}_{\beta} \underbrace{(z_1 z_2 + z_1 d - y)}_{m} \\ + \underbrace{2y - 2z_2 - 2z_1 z_2}_{\mathbf{c}(\mathbf{y})}. \end{aligned} \tag{24}$$

Since $m = 0$, we have that $c^v = c(y)$. In more complex cases, α and β have to be determined by computing a Gröbner basis for the constraint and measurement relations and by reducing c^v modulo the Gröbner basis.

4. SWITCHING OPERATING REGIONS

After the controlled variables for all regions are identified, the remaining issue is to determine how to switch between operating regions. Under certain assumptions, the switching points can be found using the already defined invariant variable combinations.

Theorem 2. (Switching regions). Assume the system (10) convex and at the optimum $\nabla_z J(\mathbf{z},\mathbf{d}) \neq 0$ wherever a constraint is active. If a disturbance moves the system continuously from one region of active constraints to another, (i.e. the system does not jump over regions) the exact switching points can be detected by monitoring the controlled variables and the constraints of the neighbouring regions.

Proof. We consider to type of changes, denoted type I and II. In changes of type I, a constraint is replaced or added to the current active set. This change is easily detected by monitoring the active constraints of the neighbouring regions. As the system is operated optimally and the disturbance moves the system gradually to the new region, the region boundary is reached, when the constraint is hit.

In changes of type II, a constraint becomes inactive and the released degree of freedom is controlled using a measurement invariant. Detecting a type II change is done by monitoring the invariant variable combinations of the neighbouring regions. If if an invariant variable combination hits the zero value, the region is switched. The invariant variable combinations assume the value zero only at the switching points. This can be seen by contradiction. Consider two regions with c_1^v and c_2^v. Let the system be operated optimally at $(\mathbf{z}_0, \mathbf{d}_0)$ in the constrained region 1 ($c_1^v(\mathbf{z}_0, \mathbf{d}_0) = 0$), and let $c_2^v(\mathbf{z}_0, \mathbf{d}_0) = 0$. The Jacobians of the set of constraints $g^1(\mathbf{z},\mathbf{d})$ and $g^2(\mathbf{z},\mathbf{d})$ are denoted as $\mathbf{A}^1(\mathbf{z},\mathbf{d})$ and $\mathbf{A}^2(\mathbf{z},\mathbf{d})$.

Since $c_1^v(\mathbf{z}_0, \mathbf{d}_0) = c_2^v(\mathbf{z}_0, \mathbf{d}_0) = 0$ we have

$$\underbrace{\left[\mathbf{N}^1(\mathbf{z}_0, \mathbf{d}_0)\right]^{\mathbf{T}} \nabla_z J(\mathbf{z}_0, \mathbf{d}_0)}_{=0} = \underbrace{\left[\mathbf{N}^2(\mathbf{z}_0, \mathbf{d}_0)\right]^{\mathbf{T}} \nabla_z J(\mathbf{z}_0, \mathbf{d}_0)}_{=0}. \tag{25}$$

Since $\nabla_z J(\mathbf{z},\mathbf{d}) \neq 0$, this implies that $\mathbf{A}^1$ and $\mathbf{A}^2$ are row equivalent and the null spaces of $\mathbf{A}^1(\mathbf{z}_0, \mathbf{d}_0)$ and $\mathbf{A}^2(\mathbf{z}_0, \mathbf{d}_0)$ have the same basis. However, this is not possible, because by assumption, $\mathbf{A}^1$ and $\mathbf{A}^2$ have different ranks. Therefore the invariant variable combination of a "less constrained" region cannot become zero in a region which is "more constrained". Thus, the active constraints and the measurement combinations can be used for determining when to switch region. □

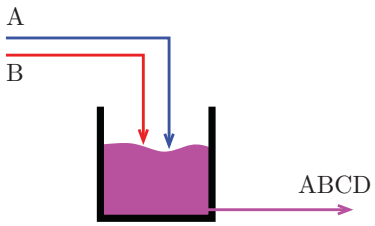


Fig. 1. CSTR with two reactions

Table 1. Variables relevant for control

Measurements $\mathbf{y}$	F, c_B, q
Manipulated variables $\mathbf{u}$	F_A, F_B
Unknown disturbance $\mathbf{d}$	Rate constant k_1
Internal states $\mathbf{z}_{unknown}$	c_A, c_C

5. APPLICATION

We consider an isothermal CSTR with two parallel reactions, Fig. 1, from Srinivasan et al. (2008). Two feed streams F_A and F_B with the concentrations c_A and c_B react in a tank to the desired product C and the undesired side product D. The tank is equipped with one outflow in which all components are present. In order to enable isothermal reaction conditions a temperature loop is closed such that the correct amount of heat is removed from the system. The temperature control is assumed to be perfect. The products C and D are formed by the reactions:

$$\begin{aligned} \mathrm{A} + \mathrm{B} &\xrightarrow{k_1} \mathrm{C} \\ 2\,\mathrm{B} &\xrightarrow{k_2} \mathrm{D} \end{aligned} \tag{26}$$

The optimization objective is to maximize the desired product $(F_A + F_B)c_C$ weighted by a yield factor $(F_A + F_B)c_C/(F_A c_{A,in})$. The amount of heat to remove and the maximum flow rate are limited. This lets us formulate the optimization problem of the system as follows:

$$\max_{F_A, F_B} \frac{(F_A + F_B)c_C}{F_A c_{A_{in}}}(F_A + F_B)c_C \tag{27}$$

subject to

$$\begin{aligned} F_A c_{A_{in}} - (F_A + F_B)c_A - k_1 c_A c_B V &= 0 \\ F_B c_{B_{in}} - (F_A + F_B)c_B - k_1 c_A c_B V - 2k_2 c_B^2 V &= 0 \\ -(F_A + F_B)c_C + k_1 c_A c_B V &= 0 \\ F_A + F_B - F &= 0 \\ k_1 c_A c_B V(-\Delta H_1) + 2k_2 c_B V(-\Delta H_2) - q &= 0 \\ q - q_{\max} &\leq 0 \\ F - F_{\max} &\leq 0 \end{aligned} \tag{28}$$

The variables k_1 and k_2 are the rate constants for the two reactions, $(-\Delta H_1)$ and $(-\Delta H_2)$ are the corresponding reaction enthalpies, q the heat produced by the reactions, V the reactor volume, and F the total flow rate. The measured variables ($\mathbf{y}$), the manipulated variables ($\mathbf{u}$), the disturbance variables ($\mathbf{d}$), and the internal states are listed in table 1, and the parameter values of the system are given in table 2. The combined vector of states and manipulated variables is

$$\mathbf{z} = [\, c_A,\ c_B,\ c_C,\ F_A,\ F_B \,]^{\mathbf{T}}. \tag{29}$$

5.1 Identifying operational regions

The first step of the procedure, optimizing the system offline for all possible values shows that the system operation space can be partitioned into three regions defined by the set of active constraints. In region 1, for values of k_1 below about $k_1 = 0.65$ only the flow constraint is active (Fig. 2). In region 2 for values between $k_1 = 0.65$ and $k_1 = 0.8$ both constraints are active, and in region 3 above $k_1 = 0.8$ only the heat constraint is active.

Table 2. Parameters

k_1	l/(mol h)	0.3-1.5
k_2	l/(mol h)	0.0014
$(-\Delta H_1)$	j/mol	7×10^4
$(-\Delta H_2)$	j/mol	5×10^4
$c_{A,in}$	mol/l	2
$c_{B,in}$	mol/l	1.5
V	l	500
$F_{\max}$	l	22
q_{max}	kJ/h	1000

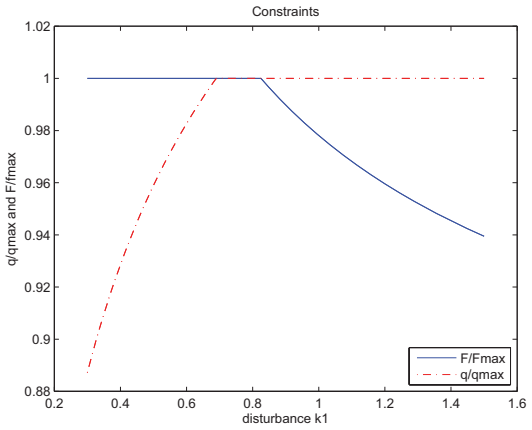


Fig. 2. Optimal values of the constrained variables

After satisfying the active constraints in the regions we are left with $N_{\mathrm{DOF},1} = 1$ for region 1, $N_{\mathrm{DOF},2} = 0$ for region 2, and $N_{\mathrm{DOF},1} = 1$ for region 3.

In region 1, one of the manipulated variables (flow rates) is used to control the active constraint (maximum flow) and the other manipulated variable is used to control the invariant measurement combination of the region. In region 2 we simply control the active constraints, keeping q at $q_{\max}$ and F at $F_{\max}$. In region 3, again one of the manipulated variables is used to control the active constraint (maximum heat removal) and the other one is used to control the invariant measurement combination of region 3.

5.2 Determining the invariant variable combinations

Using the information from the previous section, we determine the invariant variable combinations in each region. First, we calculate the null space of Jacobian of the active set $\mathbf{N}_z^{\mathbf{T}}$ and multiply it with the gradient of the objective function $\nabla_z J(\mathbf{z}, \mathbf{d})$, as in (17) to obtain the invariant variable combination. Generally this is a fractional expression, but since we are controlling it to zero, it is sufficient to consider only the numerator of $\mathbf{N}_z^{\mathbf{T}} \nabla_z J$. For *region 1* we obtain the invariant

$$\begin{aligned}\mathbf{c}_1^v(\mathbf{z},\mathbf{d}) = &-(F_A+F_B)^2 c_C \left[-3c_C F_B^2 F_A - 3c_C F_A^2 F_B\right.\\ &- 4c_C c_B F_A^2 k_2 V - 4c_C k_2 V^2 k_1 c_B^2 F_A - c_C F_A^3\\ &- c_C F_B^3 - 4c_C k_2 V^2 k_1 c_B^2 F_B - c_C c_B F_A^2 k_1 V\\ &- 4c_C c_B F_B^2 k_2 V - c_C c_B F_B^2 k_1 V - c_C F_A^2 c_A k_1 V\\ &- c_C F_B^2 c_A k_1 V - 8c_C F_A c_B F_B k_2 V\\ &- 2c_C F_A c_B F_B k_1 V - 2c_C F_A F_B c_A k_1 V\\ &+ 8F_A k_1 V^2 c_{A,in} k_2 c_B^2 + 2F_A^2 k_1 V c_B c_{A,in}\\ &+ 2F_A k_1 V F_B c_B c_{A,in} - 2F_A^2 k_1 V c_{B,in} c_A\\ &\left.-2F_A k_1 V F_B c_{B,in} c_A\right]\end{aligned} \tag{30}$$

which should be controlled to zero. This invariant can be simplified somewhat further, since we know that $(F_A + F_B)^2 c_C \neq 0$. It is therefore sufficient to control the second term in the square brackets in (30) to zero.

As mentioned above, *region 2* does not have any unconstrained degree of freedom, so satisfying all active constraints yields optimal operation. In *region 3* we obtain an expression similar to (30) for $\mathbf{c}_3^v(\mathbf{z},\mathbf{d})$.

5.3 Eliminating unknown variables

The invariant variable combination $\mathbf{c}_1^v(\mathbf{z},\mathbf{d})$ and $\mathbf{c}_3^v(\mathbf{z},\mathbf{d})$ still contain the unknown and internal variables k_1, c_a and c_C, so they cannot be used for feedback control directly. In the next step the unknown variables have to be replaced by expressions in the measured variables, so that this invariant can be used for control. Depending on the type of the system equations, different methods may be applied in this step. The general idea is that we use the measurements together with the equations that are satisfied in the active set to express the invariant. As all equations in this case study are polynomial (rational expressions equal to zero can transformed to polynomials by multiplication with the denominator), we attempt to reduce the invariants modulo the active set with a variable ordering that eliminates the unknowns. To simplify the elimination procedure, k_1 is eliminated by solving the third equality constraint for k_1.

$$k_1 = (F_A + F_B) c_C / (c_A c_B V) \tag{31}$$

and inserting it into (30). The other unknown variables c_A and c_C are eliminated using polynomial reduction and the resulting measurement invariant in *region 1* becomes:

$$\begin{aligned}c_1(\mathbf{y}) = &-F_{max}(F_{max}c_B + 2c_B^2 k_2 V - F_B c_{B,in})^2\\ &(4c_B^4 k_2^2 V^2 + 4F_{max}c_B^3 k_2 V - 6k_2 V c_B^2 F_B c_{A,in}\\ &- 4k_2 V F_{max} c_{B,in} c_B^2 + 6k_2 V c_B^2 F_{max} c_{A,in}\\ &+ F_{max}^2 c_B^2 - 2F_{max}^2 c_{B,in} c_B + 2c_B F_{max}^2 c_{A,in}\\ &- 2c_B F_{max} F_B c_{A,in} - F_B^2 c_{B,in}^2 + 3F_{max} F_B c_{A,in} c_{B,in}\\ &- F_B^2 c_{A,in} c_{B,in} + 2F_{max} F_B c_{B,in}^2 - 2F_{max}^2 c_{A,in} c_{B,in})\end{aligned} \tag{32}$$

This expression depends only on known variables and parameters. The measurement invariant for *region 3* is found in the same way:

$$\begin{aligned}c_3(\mathbf{y}) = &-(F_A c_B + c_B F_B + 2c_B^2 k_2 V - c_{B,in} F_B)^2\\ &(-3F_B^2 q_{max} c_{B,in} c_B + 8c_B^4 q_{max} k_2^2 V^2\\ &+ F_B^2 q_{max} c_{B,in}^2 + 2c_B^2 F_B^2 q_{max} + 2F_A^2 q_{max} c_B^2\\ &+ 4c_B^4 F_B k_2^2 c_{B,in} V^2 \Delta H_2 + 8c_B^3 F_B q_{max} k_2 V\\ &+ 2c_B^3 F_B^2 k_2 c_{B,in} V \Delta H_2 - 6c_B^2 F_B q_{max} k_2 c_{B,in} V\\ &- 2c_B^2 F_B^2 k_2 c_{B,in}^2 V \Delta H_2 - F_A^2 q_{max} c_{B,in} c_B\\ &+ 4F_A c_B^2 F_B q_{max} + F_A F_B q_{max} c_{B,in}^2 + 2F_A^2 c_B q_{max} c_{A,in}\\ &+ 6F_A^2 k_2 c_{B,in} V \Delta H_2 c_B^3 + 12F_A k_2^2 c_{B,in} V^2 \Delta H_2 c_B^4\\ &+ 8F_A c_B^3 F_B k_2 c_{B,in} V \Delta H_2 + 8F_A c_B^3 q_{max} k_2 V\\ &- 2F_A c_B^2 q_{max} k_2 c_{B,in} V - 6F_A c_B^2 F_B k_2 c_{B,in}^2 V \Delta H_2\\ &- 4F_A F_B q_{max} c_{B,in} c_B + 2F_A^2 c_B^2 k_2 c_{A,in} c_{B,in} V \Delta H_2\\ &- F_A^2 q_{max} c_{A,in} c_{B,in} + 4F_A^2 c_B^3 k_2 c_{A,in} V \Delta H_2\\ &+ 4F_A c_B^3 F_B k_2 c_{A,in} V \Delta H_2 - 2F_A c_B^2 F_B k_2 c_{A,in} c_{B,in} V \Delta H_2\\ &+ 4F_A c_B^2 q_{max} k_2 c_{A,in} V + 2F_A c_B F_B q_{max} c_{A,in}\\ &- F_A F_B q_{max} c_{A,in} c_{B,in})\end{aligned} \tag{33}$$

Although these expressions seem complicated, they contain only known variables and can therefore be easily evaluated and controlled to their setpoint using a PI controller. In both invariants, the term in the first bracket is never zero (to see this, compare it with the second equality constraint in (28)), so it is sufficient to control the term in the second bracket to zero.

The values of these polynomials vary over order of magnitudes, so they are scaled to avoid numerical problems. The invariant of region 1 was scaled by 10^5 and the invariant of region 3 was scaled by 10^6.

5.4 Using measurement invariants for control and region identification

We use the controlled variables of the neighbouring regions for determining when to switch. Starting in region 1 optimal operation is achieved by using the two inputs F_A and F_B to control $c_1(\mathbf{y}) = 0$ and $F_A + F_B = F_{max}$. If k_1 increases, the amount of heat to be removed (the controlled variable of region 2) increases until it reaches the maximum possible value, q_{max} (Fig 3). When this value is reached, the control structure has to be changed to region 2. Now the inputs are used to control $q = q_{max}$ and $F_A + F_B = F_{max}$. While operating in region 2 the controlled variables of the neighbouring regions, $c_1(\mathbf{y})$ and $c_3(\mathbf{y})$ are monitored. If k_1 increases further, $c_3(\mathbf{y})$ approaches its optimal setpoint for region 3 and we switch region when the optimal value is reached. Switching back from the different regions is done in an analog manner.

6. DISCUSSION

The invariant variable combinations above are obtained by a two-step method, in which first the Lagrangian multipliers are eliminated, and subsequently the unknown variables (disturbances and internal states) are replaced by measurement relations. However, for systems which can be described by rational functions, as the CSTR example, there exists possibilities to eliminate both, the Lagrangian multipliers and the unknown variables simultaneously.

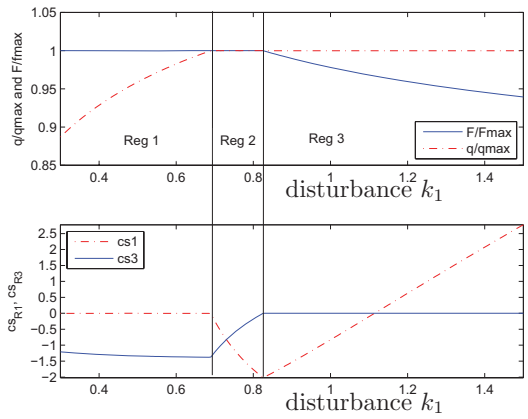


Fig. 3. Optimal values of controlled variables

This is done by defining an ideal which is generated by the polynomials describing the Karush-Kuhn-Tucker conditions and choosing an appropriate term order to eliminate the unknown disturbance and state variables (Cox et al. (1992)) Applying the chosen order to generate the elimination ideal gives a set of equations which are fulfilled at all times when the KKT conditions hold, but which do not contain any of the unknown variables.

Choosing a polynomial from this set which is not a (polynomial) combination of the equality constraints (i.e. which is not in the ideal generated by the equality constraints), gives a candidate for the measurement invariant. However, presently there are two challenges with the simultaneous method. First, the system with the chosen invariant measurement combinations may have (many) more roots than the KKT system. If we use them for control, we might control to "solutions" which do not satisfy the first order optimality conditions. Second, it is generally difficult to determine a term ordering a priori which eliminates the unknown variables and, at the same time ensures that the equations in the elimination ideal are not polynomial combinations of the equality constraints. If this is the case, the chosen variable combination is in the ideal generated by the equality constraints and the invariant will always be zero when the equality constraints are satisfied. This leads to an infinite number of solutions for the system. So far we are not aware of a method to handle these challenges in a systematic way, but we have found invariants in the elimination ideals which yield optimal operation in for the CSTR case study shown above.

The two step method presented in this work fundamentally shows the existence of invariant variable combinations for nonlinear systems and gives an easy way to compute and to use them. Additionally, in contrast to the elimination ideal method, the two-step method is principally not restricted to polynomial or rational models, provided that the unknowns can be eliminated.

7. CONCLUSION

The procedure presented in this paper is applicable to nonlinear steady state optimization problems and consists of four steps. First, regions of constant active constraints are defined. Second, optimally invariant nonlinear variable combinations are determined for each of the regions. Third, the unknown internal variables and disturbances are eliminated from the invariants to obtain variable combinations containing only known variables (measurements). It is proven that these variables can be used to uniquely identify a new active set. This makes the method applicable over a wide disturbance range with changing active sets. Finally, we have applied the method to a case study with a four component isothermal CSTR.

Although designing a self-optimizing control structure may require more work in advance, its implementation and maintenance is easy in practice. After the control structure is designed, optimal operation can achieved by simple PI controllers and there is no need to invest in expensive real-time equipment to operate the process optimally.

REFERENCES

Alstad, V., Skogestad, S., and Hori, E. (2009). Optimal measurement combinations as controlled variables. *Journal of Process Control*.

Cox, D., Little, J., and O'Shea, D. (1992). *Ideals, Varieties, and Algorithms*. Springer-Verlag.

Manum, H., Narasimhan, S., and Skogestad, S. (2007). A new approach to explicit mpc using self-optimizing control, *Internal report*. `www.nt.ntnu.no/users/skoge/publications/2007`.

Narasimhan, S. and Skogestad, S. (2007). Implementation of optimal operations using off-line computations. *8th international IFAC Symposium on Dynamics and Control of Process Systems (DYCOPS), Cancun, Mexico*.

Nocedal, J. and Wright, S.J. (2006). *Numerical Optimization*. Springer.

Skogestad, S. (2000). Plantwide control: The search for the self-optimizing control structure. *Journal of Process Control*, 10, 487–507.

Srinivasan, B., Biegler, L.T., and Bonvin, D. (2008). Tracking the necessary conditions of optimality with changing set of active constraints using a barrier-penalty function. *Computers and Chemical Engineering*, 32, 572–279.

Appendix A. ELIMINATING INTERNAL STATES

We define the combined input and internal state vector as

$$\mathbf{z} = \begin{bmatrix} \mathbf{x}^\mathbf{T} & \mathbf{u}^\mathbf{T} \end{bmatrix}^\mathbf{T}. \tag{A.1}$$

It is assumed that we have $n_u + n_d$ independent measurements for system 3.

$$\mathbf{y} = \tilde{\mathbf{G}}^y \begin{bmatrix} \mathbf{u}^\mathbf{T} & \mathbf{d}^\mathbf{T} \end{bmatrix} \tag{A.2}$$

with $\tilde{\mathbf{G}}^y$ invertible. Then

$$\begin{bmatrix} \mathbf{z} \\ \mathbf{d} \end{bmatrix} = \begin{bmatrix} \mathbf{x} \\ \mathbf{u} \\ \mathbf{d} \end{bmatrix} = \begin{bmatrix} -\mathbf{A}_x^{-1}\mathbf{A}_u & -\mathbf{A}_x^{-1}\mathbf{A}_s \\ \mathbf{I} & 0 \\ 0 & \mathbf{I} \end{bmatrix} \underbrace{\begin{bmatrix} \mathbf{u} \\ \mathbf{d} \end{bmatrix}}_{=[\tilde{\mathbf{G}}^y]^{-1}\mathbf{y}}, \tag{A.3}$$

and we derive the optimal measurement combination which satisfies (7) as

$$\mathbf{c}(\mathbf{y}) = \mathbf{H}\mathbf{y} \tag{A.4}$$

with

$$\mathbf{H} = \mathbf{N}_z^\mathbf{T} \tilde{J} \begin{bmatrix} -\mathbf{A}_x^{-1}\mathbf{A}_u & -\mathbf{A}_x^{-1}\mathbf{A}_s \\ \mathbf{I} & 0 \\ 0 & \mathbf{I} \end{bmatrix} \left[\tilde{\mathbf{G}}^y\right]^{-1}. \tag{A.5}$$

Influence of Differences in System Dynamics in the context of Multi-unit Optimization

François Reney, Michel Perrier, Bala Srinivasan.

Chemical Engineering Department, École Polytechnique, Montreal, Quebec, Canada (e-mail: bala.srinivasan@polymtl.ca).

Abstract: Extremum-seeking methods are unconstrained real-time optimization techniques that control the gradient to zero. The crucial difference between them lies in the gradient estimation method used. Multi-unit optimization technique proposes the use of a multiple units operated with an offset between them and the estimation of the gradient is by finite difference. Though this method gives fast convergence, the major bottleneck is that it assumes the units to be identical. This paper addresses the case where the static curves are indeed identical, while the dynamics are not so. It is shown that if all the units are stable, despite the difference in dynamics, the method would indeed converge to the true optimum. Also, it is shown that the difference in dynamics does not affect stability in the neighborhood of the optimum. In addition, this paper presents a possibility of replacing real units by static models in the calculation of the gradient. Experimental results are presented from a mixing system where an optimal temperature is sought.

Keywords: Real time optimization and control, multi-units optimization.

1. INTRODUCTION

Process optimization is a tool of choice to find the best operating point that balances conflicting objectives such as productivity, selectivity, and operating cost for continuous chemical process. To perform this optimization numerically, it is necessary to have a model of its operation. Though most processes are dynamic in nature, often a steady state model suffices since typically, for continuous process, one is interested in finding the best steady state operation point. However, due to process changes, the optimal operating point varies with time, and to reap the benefits, it is indeed crucial to track these changes.

Without being exhaustive, two main classes of techniques have been employed in the real-time optimization of continuous processes. The first class comprises of the repeated optimization techniques (Marlin & Hrymak, 2000) that alternate between the identification of a steady state model using measurements and numerical computation of the optimal input using the updated model. On the other hand, extremum-seeking methods (Ariyur & Krstic, 2003, Guay et al., 2004) treat the optimization problem as one of controlling the gradient to zero.

In the extremum-seeking framework, various methods have been used for the gradient determination. The perturbation method (Ariyur & Krstic, 2003) deduces the gradient by adding perturbation signal that is very slow compared to the process dynamics. The correlation between the input and output is used to estimate the gradient. In adaptive extremum seeking techniques (Guay et al., 2004), parameters of a dynamic model are adapted and the gradient is computed from the adapted model. All the above mentioned techniques have to respect time-scale separations between gradient estimation and the process dynamics, thereby leading to slow convergence.

The multi-unit optimization technique (Srinivasan, 2007) is an attempt to find another gradient method that would converge faster. The basic method uses two identical units operated with an offset between them and uses finite difference to estimate the gradient. However, the main drawback of this technique is that it needs multiple identical processes working in parallel, which is impossible to get in practice. It has been shown that, if the units are not identical, the stability of the scheme and the convergence toward the true optimum are not guaranteed (Woodward et al., 2009).

Further research (Woodward et al., 2009) has revealed a way to compensate the difference in the curves that represent the steady-state relationship between the input and the objective function. It uses translation in both the input direction and direction of the objective function so as to evaluate correctly the gradient and converge toward the true optimum.

The key advantage of multi-unit optimization technique is that a reliable gradient is available during the transient, and one need not have to wait for the steady state. This advantage arises from the fact that if the process dynamics are the same, the difference of the objective functions is rendered insensitive to the process dynamics. However, if the processes' dynamics are different, even if the static curves are identical, the gradient would be falsified. The stability and convergence to the true optimum are a priori not assured.

In this paper, the case of non-identical dynamics is considered. It is however assumed that the static curves are the same. Such a case occurs when the optimization objective is only a function of the system output whose dynamics is controlled by a controller with integrator. Results from

stability analysis and the equilibrium point are presented. It is shown that the difference in dynamics does not change the equilibrium point but it indeed affects stability and the way it converges. The theoretical results are experimentally verified and the data are also presented in this paper.

This paper also analyses the possibility of replacing a physical unit by a mathematical model. This way, the multi-unit optimization runs with one physical system and one mathematical model. Here, experimental results with a physical dynamic system and a static mathematical model are presented to show that this option is indeed viable.

The rest of this paper is organized as follows. Section 2 of this paper presents briefly the standard multi-unit optimization technique. Section 3 presents the results of the analysis for stability and convergence when the units' dynamic are different. Section 4 presents the methodology and the results for the experimental trials.

2. MULTI-UNIT OPTIMIZATION

2.1 Problem formulation

Mathematically, a standard real time optimization problem is written as follows:

$$\begin{aligned} &\min_{u} \quad J = g(x,u) \\ &\dot{x} = f(x,u) = 0 \end{aligned} \tag{1}$$

J is a twice-differentiable function that is minimized and f represents the dynamics of a stable process. The states of the system are represented by the vector *x* and the inputs by the vector *u*. For the easing of presentation, inequality constraints are ignored.

In order to find the optimal input, it is easier to use the equality constraints to find an expression of x = h(u) and then substitute the same. This transforms the original problem into a unconstrained optimization problem, i.e. min J = p(u). Then, the necessary condition of optimality is then given by:

$$\partial p/\partial u \,|_{u^*} = 0 \tag{2}$$

If it is assumed that the unconstrained optimization problem is convex, then the necessary condition indeed leads to the only minimum. Equation (2) is used in extremum-seeking methods to find the optimal point by gradient control.

2.2 Multi-unit optimization scheme

A schematic representation of a simplified version of the multi-unit optimization framework is shown in Fig. 1 (Srinivasan, 2007). The term "unit" is used to represent a real continuous chemical process, and here they are labeled "0" and "1". A difference of Δ between the inputs "u_0" and "u_1" is necessary to estimate the gradient by a first order finite difference equation $\partial J/\partial u = (J_1 - J_0)/\Delta$. Then, the method uses an integral controller with an appropriate value of gain to push the gradient towards 0. The gain K can be tuned to as a compromise between convergence speed and stability.

$$\dot{u}_0 = \dot{u}_1 = \frac{K}{\Delta}\left(J_1 - J_0\right) \tag{3a}$$

$$u_0 = u - \frac{\Delta}{2} \text{ and } u_1 = u + \frac{\Delta}{2} \tag{3b}$$

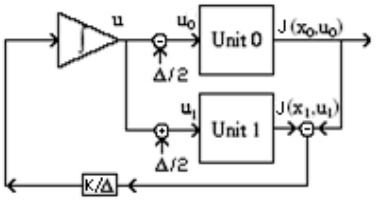


Fig. 1. Standard multi-unit optimization control loop.

In the case of multi-unit optimization, it is necessary to precise the dynamics of each of the units. The two units can be written mathematically as follows:

$$\begin{aligned} \dot{x}_0 &= f_0(x_0,u_0) \\ \dot{x}_1 &= f_1(x_1,u_1) \\ J_0 &= g_0(x_0,u_0) \\ J_1 &= g_1(x_1,u_1) \end{aligned} \tag{4}$$

The steady state for each unit can be described by:

$$\begin{aligned} x_0 &= h_0(u_0) \\ x_1 &= h_1(u_1) \end{aligned} \tag{5}$$

And the objective functions of each of the units at steady-state given by:

$$\begin{aligned} J_0 &= g_0(h_0(u_0),u_0) = p_0(u_0) \\ J_1 &= g_1(h_1(u_1),u_1) = p_1(u_1) \end{aligned} \tag{6}$$

3. MULTI-UNIT OPTIMIZATION WITH DIFFERENT DYNAMICS

In this section, the results of the analyses for stability and convergence are presented for the case when the units' dynamic are different but the static curve for each unit are identical. These analyses are made assuming that the technique is applied without any modification to compensate for the difference in dynamics.

3.1 Analysis for the equilibrium point

Here it is shown that if the scheme is stable, the system will converge toward the true optimum as long as the static curves between the units are identical. The difference between the dynamics does not bias the equilibrium point.

Theorem 1: If (i) the scheme converges, (ii) the static curves of the two units are identical, then despite the difference in the dynamics, the steady state of the multi-units optimization control loop represents the real optimum as Δ tends to zero.

Proof: From (3a), it can be seen that at steady state:

$$\dot{u}_0 = \dot{u}_1 = 0 = \frac{K}{\Delta}\left(J_1 - J_0\right) = \frac{K}{\Delta}\left(p_1(u_1) - p_0(u_0)\right) \tag{7}$$

So, the equilibrium point is determined by:

$$p_0(u_0) = p_1(u_1) \tag{8}$$

As the static curves of two units are considered identical,

$$p_0(u) = p_1(u) = p(u) \tag{9}$$

Let $\bar{u}$ be the value of u at steady state. Then,

$$p\left(\bar{u} + \Delta/2\right) = p\left(\bar{u} - \Delta/2\right) \tag{11}$$

A second order Taylor expansion of the function p is considered:

$$p(\bar{u}) + p'(\bar{u})\frac{\Delta}{2} + p''(\bar{u})\frac{\Delta^2}{4} - p(\bar{u}) + p'(\bar{u})\frac{\Delta}{2} - p''(\bar{u})\frac{\Delta^2}{4} + O(\Delta^3) = 0 \tag{12}$$

$$p'(\bar{u}) = O(\Delta^2) \tag{13}$$

$$\lim_{\Delta \to 0} p'(\bar{u}) = \lim_{\Delta \to 0} O(\Delta^2) = 0 \tag{14}$$

Calculating the limit as Δ tends to zero, it is observed that the derivative becomes zero, indicating that the two units arrive at optimal point. ■

3.2 Analysis for the stability of the scheme.

To analyze the stability of the above scheme, the units are linearized around the current operating point. The transfer function representation is used and normalized transfer function that has a unit steady state gain is derived. Then, the characteristic equation of the loop is obtained and analyzed if the roots of this equation are in the left half of the complex plane.

Theorem 2: If an integral controller with gain K can stabilize the average of the two normalized dynamics, then for a small enough value of Δ, the scheme is locally asymptotically stable around the optimum.

Proof: In order to analyze locally the stability, it is necessary to linearize the dynamics of both units. With the linearization, it is possible to represent directly the relationship between u and J by a transfer function T(s). It is useful to rewrite T(s) as a combination of a dynamic term with a static gain of 1, labeled as N(s), and a static term which represents the steady state gain:

$$T_0(s) = N_0(s)p'(u_0) \text{ and } T_1(s) = N_1(s)p'(u_1) \tag{15}$$

Note that the static gain of the different units is given by the gradient (linearization) of the static curve p(u) at their respective operating points. The above decomposition separates the static behavior of the units from its dynamics. The condition on the characteristic equation for this loop to be stable is given by:

$$Z = 1 + \frac{K}{s\Delta}\left(T_1(s) - T_0(s)\right) = 0 \tag{16}$$

Considering a Taylor expansion approximation of p gives:

$$p'(u_0) = b - \frac{\Delta}{2}M \text{ and } p'(u_1) = b + \frac{\Delta}{2}M \tag{17}$$

where b = p'(u) and M = p"(u). Distributing and rearranging the terms in (16), one gets,

$$s + \frac{KM}{2}\left(N_1 + N_0\right) + \frac{Kb}{\Delta}\left(N_1 - N_0\right) = 0 \tag{18}$$

At the equilibrium point (which has been shown to be the optimum in Theorem 1), as Δ tends to zero, (b/Δ) goes to zero. Then, the third term of the characteristic equation disappears. So, around the optimum, the stability is no more influenced by the value of Δ, nor by the error in the dynamics. The stability of the system around the optimum is then determined by whether or not the integral controller with the given gain stabilizes the average dynamics. ■

Note that the difference in the dynamics would not affect the stability around the optimum. However, the difference in dynamics can affect the characteristic equation considerably when the system is far from the optimum. The characteristic equation (18) is a rich source of information from which two conclusions can be drawn:

Effect of gain K: The value of the gain is crucial to stability in all cases. If the normalized average dynamics is stable and minimum phase, then for small values of gain, the overall scheme would be stable. Moreover, if the value of K is small, it can be seen from (18) that the influence of the difference in dynamics is negligible. In other words, with a small gain, the inputs variations are slow compared to the dynamics, and the two units operate at their respective pseudo-steady states.

Evolution far from the optimum: When the starting point is far from optimum ("b" is not zero), it can be seen from (18) that the dynamic behavior is influenced by the sign of the ratio (b/Δ), rather than the sign of the individual components.

Without loss of generality, suppose unit 0 is faster than unit 1. A negative Δ and a negative b mean that the faster unit is closer to the optimum. The same situation occurs for a positive Δ and a positive b. These two cases would have similar adaptation characteristics, and are shown in Figure 3 with the red dot representing the faster unit 0 and the green representing the slower unit 1.

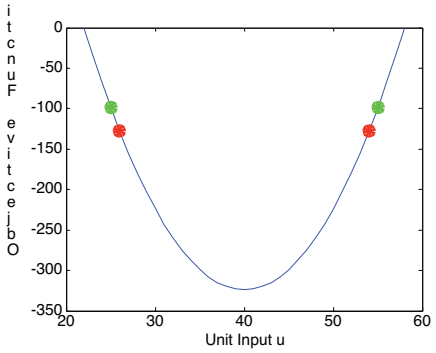


Fig. 3. Analysis of configurations with respect to dynamic behavior. Red faster than green leads to oscillations. Green faster than red leads to slow convergence.

In this configuration, since the system with a rapid dynamics is closer to the optimum, the difference between the outputs will be larger during the transients (for the same change in u). This will overestimate the gradient which, in the best case, will cause the system to converge with oscillations.

A negative Δ and positive b means that the slower unit is closer to the optimum. The same situation occurs for a positive Δ and a negative b. These two cases would have

similar dynamics, and are shown in Figure 3 with the green dot representing the faster unit 0 and the red representing the slower unit 1.

In this configuration, since the system with a slower dynamics is closer to the optimum, the difference between the outputs will be smaller during the transients (for the same change in u). This will underestimate the gradient which will cause the system to converge slowly.

The interesting point is that even if there is an overestimation of the gradient (faster unit closer to the optimum), the system would not in general become unstable. This is due to the fact that once both the units overshoot the optimum, the situation is reversed, i.e., the slower unit is closed to the optimum. So, an under estimation of the gradient occurs, where the return back would be slow and sure.

4. EXPERIMENTAL RESULTS

4.1 Problem Formulation

In order to prove the theoretical results shown in the previous section, an experimental setup has been designed. The setup is composed of two tanks (units) whose temperature is controlled in order to minimize the criterion mentioned below.

$$\min_{F_c} \quad J = (T_{out} - T_c)(T_{out} - T_h)$$
$$\dot{T}_{out} = \frac{F_h}{V}(T_h - T_{out}) + \frac{F_c}{V}(T_c - T_{out}) = 0 \tag{19}$$

Each unit is supplied with water by two pumps: a hot water pump and cold water pump. Hot water at temperature T_h and cold water at temperature T_c are added to these tanks with flow rates F_h and F_c respectively. The hot water pump is fixed with two heads in order to feed the same flow for both units. However, each unit has its own cold water pump (F_{c0} and F_{c1} being the decision variable). The temperatures in the units T_{out0} and T_{out1}, in the hot water tank and in the cold water tank are measured using thermistors. V is the volume of each of the units. The answer to this problem is:

$$T_{out} = \frac{T_h + T_c}{2}; \quad F_c = F_h \tag{20}$$

Also, in order to control the dynamics of each unit independently, cascade control has been implemented. Essentially, the multi-unit scheme sends a temperature set point u to each temperature control loop, which is controlled using a PI controller. This way that the static curves are identical (steady state error is zero). However, by tuning the temperature controllers differently, the two units will have different dynamics.

4.2 Experiments conditions tested

Four experiments are performed for all cases. In Experiment 1, the system is initialized to a value higher than the optimum with a positive value of Δ. Experiment 2 starts with an initial condition that is less than the optimum. Experiment 3 and Experiment 4 start from the initial conditions of Experiment 1 and 2 respectively, but with negative Δ.

It is always arranged that Unit 1 is slower than Unit 0. So, in Experiments 1 and 4, the optimum is closer to the faster unit, while the optimum is closer to the slower unit in the other two experiments. Experiments 1 and 4 show a configuration that overestimates the gradient, while experiments 2 and 3 present a configuration that underestimates the gradient. The gain K is the same for all experiment and is chosen so that the scheme is always stable.

Exp.	Start Temp	Temp Unit 0 (fast)	Temp Unit 1 (slow)
1	47 C	+1 C	-1 C
2	33 C	+1 C	-1 C
3	47 C	-1 C	+1 C
4	33 C	-1 C	+1 C

Table 1: Experimental plan for all experiments

4.3 Experiments with two real units.

In this set of experiments, the controller of "unit 1" has been tuned so that the internal loop (dynamics between the temperature output and temperature set point) is two times slower than its counterpart "unit 0". The results are presented in Figures 5-8.

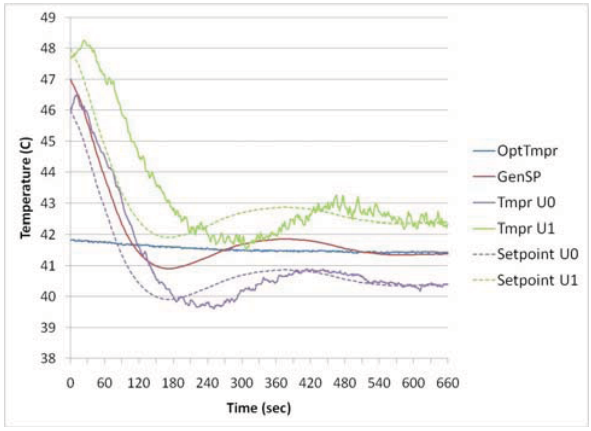


Fig. 5. Evolution for Experiment 1 with real units.

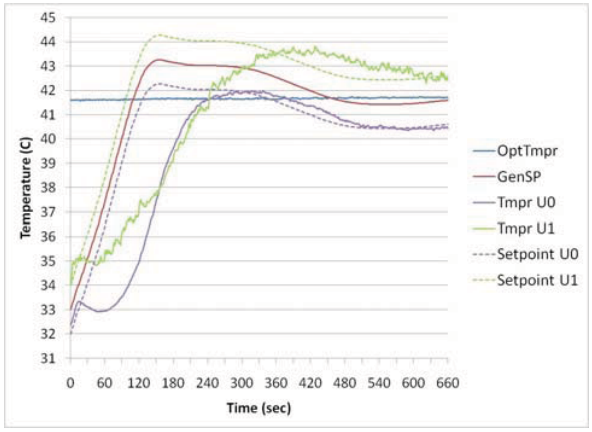


Fig. 6. Evolution for Experiment 2 with real units.

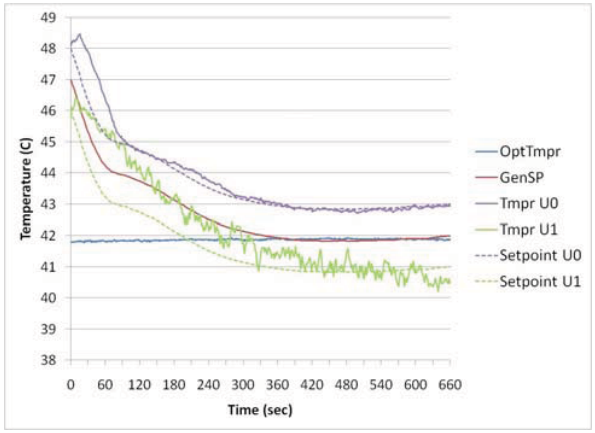


Fig. 7. Evolution of the system for Exp. 3 with real units.

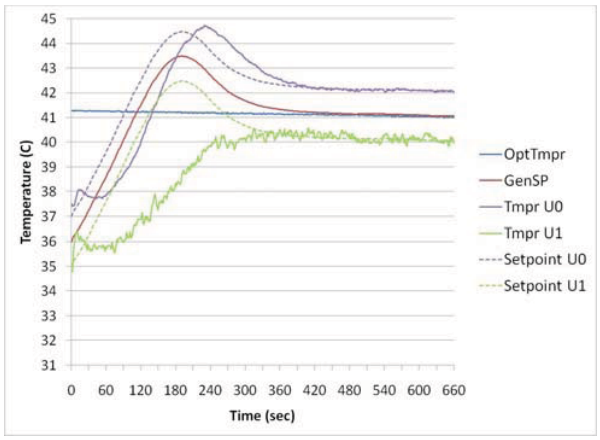


Fig. 8. Evolution of the system for Exp. 4 with real units.

The influence of the sign of Δ can be seen by comparing Fig 5 (Exp1) and Fig 7 (Exp3). The positive value of Δ brings the faster unit closer to the optimum in Exp1 and farther from the optimum in Exp3. Note that in both these experiments, there is a larger difference between the set point and the measured temperature in Unit 1 compared to that of Unit 0. The effect of the transients is more marked when the system is far from the optimum where the set point changes rapidly.

In the case of Exp 1, because of the difference in speed of the respective responses, the gradient is overestimated. This causes the scheme to converge with oscillations. On the contrary, in the case of Exp 3, the gradient is underestimated due the above mentioned speed difference. This underestimation causes the scheme to converge slowly but surely.

The above conclusion can be generalized for the ratio (b/Δ). Comparing Fig 5 (Exp1) and Fig 8 (Exp4), it can be seen that in both these experiments, the faster unit is closer to the optimum ($b/\Delta > 0$). The gradient is overestimated, resulting in an oscillatory response toward the optimum. In contrast, Fig 6 (Exp2) and Fig 7 (Exp3) show slower convergence since the faster unit is farther from the optimum ($b/\Delta < 0$).

Comparing Exp 1 and Exp3, it can be noted that the response for Exp3 is not as smooth as expected. Taking into account that a linear controller was used to control a nonlinear system, such a behavior is expected, The controller parameters are clearly not tuned for all points of operation. The controller settings used favored higher temperatures as can be seen from Figure 3.

4.3 Experiments with a real unit and a virtual unit

The purpose of next series of tests is to check the viability of replacing one of the real units by a mathematical model. This removes one of the major constraints of the scheme, i.e., the availability of two physical identical units in operation. There are two kinds of model that one can use: (i) a first principles/black box dynamic model (ii) a black box static model. Though the dynamic models were tested experimentally, the more interesting and extreme case, i.e., the use of the static model, is presented here.

In this set of experiments, “unit 0” is a static mathematical model whose dynamics is by definition instantaneous and so faster compared to the real unit (unit 1). In practice, it is done by setting the output equal to the set point of the control loop.

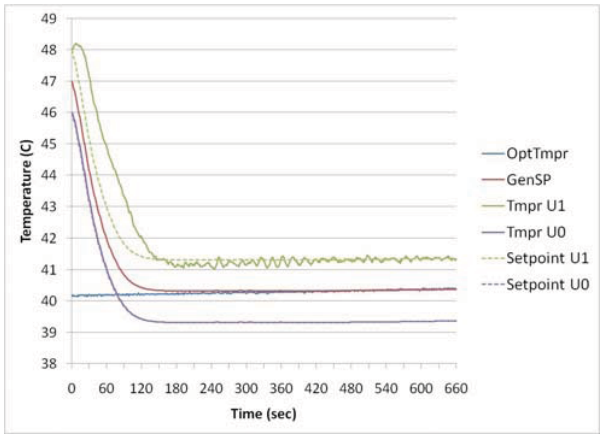


Fig. 9. Evolution for Exp. 1v with real and virtual units.

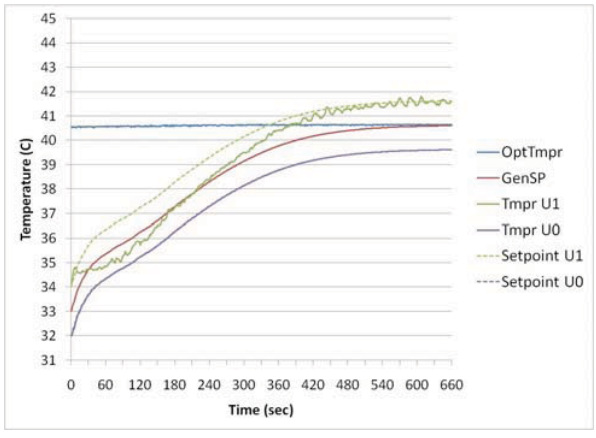


Fig. 10. Evolution for Exp. 2v with real and virtual units.

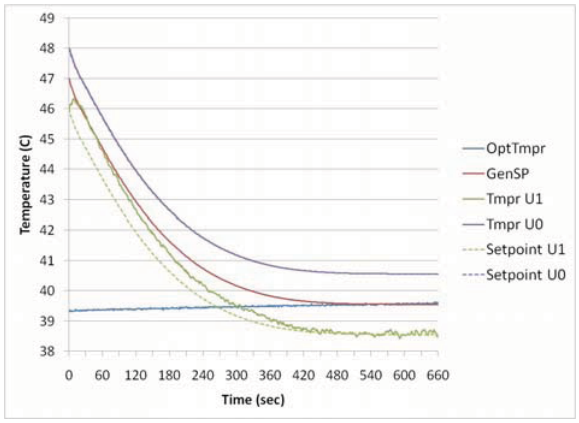


Fig. 11. Evolution for Exp. 3v with real and virtual units.

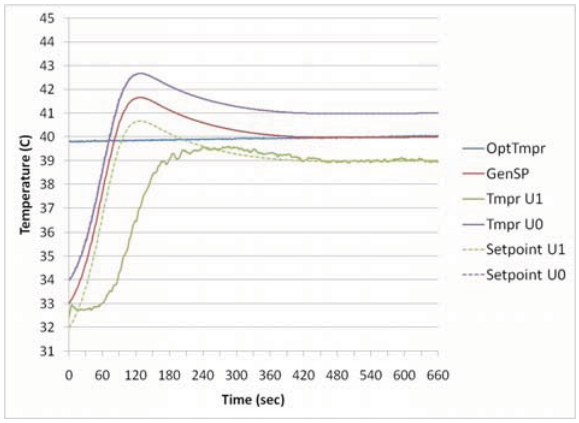


Fig. 12. Evolution for Exp. 4v with real and virtual units.

The results obtained using a system with a real unit and a virtual unit are similar to those obtained with two real units under the same testing conditions. The conditions on stability and convergence remain the same for this series of experiments. Indeed, the comparison of Fig 9 (Exp1v) and Fig 11 (Exp3v) yields the same general conclusion as the comparison of Exp1 and Exp3. It is also true for the comparison of Fig 9 (Exp1v) and Fig 12 (Exp4v) with Exp1 and Exp4 and the comparison of Fig 10 (Exp2v) and Fig 11 (Exp3v) with Exp2 and Exp3. However, for the last comparison, it is easier now to see that the smaller difference between the unit temperatures results in an underestimation of the gradient, thereby leading to slow convergence.

Note that while replacing the real unit by a model, it was assumed that the static characteristics are matched. The only difference between the real and virtual units is at the dynamics level. In this particular example, since the objective function is only dependant on the unit temperature, this assumption of matching the static behavior is easily verified. However, if the objective function is a function of both the output and input (T_{out} and F_c) of the dynamic part, it then becomes mandatory to have a good model of the physical system in order to match the static characteristics.

The conclusion here is that if the static characteristics are matched, then multi-unit optimization can be performed even when the dynamics are not necessarily identical. In extension, a real unit can be replaced by a virtual static model, which has the same static characteristics as the real unit. However, in order to converge to the true optimum, where the static characteristics are different, parameter adaptations are indeed necessary so as to compensate for these differences. In short, the differences in the dynamics can be tolerated, while differences in the static behavior needs to be quantified and compensated.

5. CONCLUSIONS

In this paper, it is shown that it possible to use the multi-unit scheme with differences in dynamics without affect its performance considerably. If the static curves are the same, the equilibrium point and the stability around the optimum are not affected. However, far from the optimum, the choice of the offset plays an important role; it can either make the system converge slowly or make it oscillatory.

Experimentally, it is shown that replacing a real dynamic unit by a simple static mathematical model is indeed viable. This means that the major constraint of having real multiple identical units can be circumvented. A good, not necessarily perfect, approximation of the process is sufficient to this effect.

REFERENCES

Srinivasan B. (2007). Real-time optimization of dynamic systems using multiple units. International Journal of Robust and Nonlinear Control 17, 1183–1193.

Woodward L., Perrier M., Srinivasan B. (2009). Convergence of Multi-Unit optimization with non-identical units: Application to the optimization of a Bioreactor. Journal of Process control, 19(2), 205-215.

Reney F. (2008). Étude de la technique d'optimisation multi-unités avec dynamiques différentes. Master's thesis at École Polytechnique de Montréal.

Guay M, Dochain D, Perrier M. (2004). Adaptive extremum seeking control of continuous stirred tank bioreactors with unknown growth kinetics. Automatica, 40 (5), 881–888

Ariyur K, Kristic M. (2003). Real-time Optimization by Extremum-Seeking Control, John Wiley and Sons.

Marlin T, Hrymak A. (1997). Real-time operations optimization of continuous processes. Vol. 316 of AIChE Symposium Series, p. 156.

A Model-Free Methodology for the Optimization of Batch Processes: Design of Dynamic Experiments

Christos Georgakis*

**Department of Chemical and Biological Engineering and Systems Research Institute*
Tufts University, Medford MA 02155, USA
(Tel: +1-617-627-2573; e-mail: Christos.Georgakis@Tufts.edu).

Abstract: The new methodology presented provides a way to optimize the operation of a variety of batch processes (chemical, pharmaceutical, food processing, etc.) especially when at least one time-varying operating decision function needs to be selected. This methodology calculates the optimal operation without the use of an *a priori* model that describes in some accuracy the internal process characteristics. The approach generalizes the classical and widely used Design of Experiments (DoE), which is limited in its consideration of decision variables that are constant with time. The new approach, called the Design of Dynamic Experiments (DoDE), systematically designs experiments that explore a considerable number of dynamic signatures in the time variation of the unknown decision function(s). Constrained optimization of the interpolated response surface model, calculated from the performance of the experiments, leads to the selection of the optimal operating conditions. Two examples demonstrate the powerful utility of the method. The first examines a simple reversible reaction in a batch reactor, where the time-dependant reactor temperature is the decision function. The second example examines the optimization of a penicillin fermentation process, where the feeding profile of the substrate is the decision variable. In both cases, a finite number of experiments (4 or 16, respectively) lead to the very quick and efficient optimization of the process.

Keywords: Batch Optimization, Design of Experiments, Batch Reactors, Fermentation, Penicillin Production, Batch Modeling.

1. INTRODUCTION

Batch processes are often related to small production rates resulting in processes that are not understood enough to enable the development of an accurate mathematical model describing their inner workings. To accommodate such a lack of detailed understanding, our research group introduced the concept of Tendency Modelling (Fotopoulos, Georgakis, & Stenger, 1996, 1998) which has been applied to several processes with significant success. See for example (Cabassud et al., 2005; Martinez, 2005). On the other hand, François et al. (François, Srinivasan, & Bonvin, 2005) have also introduced a methodology in which the feedback control concept is used to evolve from an initial batch operation to operations that are incrementally better and, after several cycles, arrive at an optimum operation. In the case that a model is available, several model-based optimization techniques can be utilized (Biegler, 2007). We will refer to this model-based approach as the *Classical Approach*.

2. THE CLASSICAL MODEL-BASED APPROACH

The classical approach in optimizing a batch process assumes we have a first-principles model describing our fundamental understanding of the process. Assuming that all important idiosyncrasies of the process are known to make the model quite comprehensive and accurate, one needs only to account for the model's parameters whose values are not well known. Based on the number of unknown parameters, a set of experiments is designed using the classical Design of Experiments (DoE) approach (Box & Draper, 2007; Montgomery, 2005), or any other systematic or not so systematic approach. Once the experimental data are collected, the model parameters can be calculated using a parameter estimation method and related algorithms (van den Bos, 2007). Such a model will often have the form of a set of nonlinear ordinary differential equations (ODEs), as in eq. 1.

$$\frac{d\mathbf{x}}{dt} = \mathbf{f}(\mathbf{x},\mathbf{p},\mathbf{u},t), \quad \mathbf{y}(t) = \mathbf{g}(\mathbf{x}); \quad \text{with } \mathbf{x}(0)=\mathbf{x}_0 \quad \text{and } \mathbf{u}_{\min} \le \mathbf{u}(t) \le \mathbf{u}_{\max} \tag{1}$$

Here, **x** and **y** represent the states and output variables of the system, respectively; **p** the parameters of the model fitted to the experiments; and **u**(t) the decision variable with which we wish to maximize (or minimize) the system's performance index *J*. The performance index is assumed to be only a function of the final values of the state variable at the end of the batch at $t=t_B$:

$$J^* = \max_{\mathbf{u}(t)} J(\mathbf{x}(t_B)) \tag{2}$$

With such a model at hand, one can calculate the optimum value of the decision variable **u**(t) that will yield the optimum value J^* of the performance index *J*. There are

several ways such a calculation can be performed, but here we will follow the method strongly advocated by Professor Biegler's group (Biegler, 2007; Kameswaran & Biegler, 2006, 2008). In such an approach, the interval (0, t_B) is divided into a number of finite elements and inside each element, the method of orthogonal collocations (Biegler, 1984) is used to convert the set of ODEs into a set of algebraic equations. Then, an optimization algorithm, such as sequential quadratic programming, calculates the optimum.

In summary, the Classical Model-based Optimization (CMO) approach involves the following steps: i) Postulation of model, ii) Experiments, iii) Parameter Estimation, and iv) Optimization.

3. THE NEW APPROACH: DESIGN OF DYNAMIC EXPERIMENTS

3.1. The Main Idea

To facilitate the discussion that follows, let us define a dimensionless time τ equal to t/t_B. The decision variable $\mathbf{u}(\tau)$ is considered to be a member of the Hilbert space $\mathcal{L}_2(0,1)$ of square-integrable vector functions. Let us denote with $\{\varphi_i(\tau);\ i = 1, 2, 3, \ldots\}$ a convenient set of basis-functions in that space. The unknown function $\mathbf{u}(\tau)$ can be written as follows.

$$\mathbf{u}(\tau) = \mathbf{u}_0 + \Delta\mathbf{U}\left(\sum_{i=1}^{\infty}\mathbf{a}_i\phi_i(\tau)\right) \simeq \mathbf{u}_0 + \Delta\mathbf{U}\left(\sum_{i=1}^{N}\mathbf{a}_i\phi_i(\tau)\right) \qquad (3)$$
$$\mathbf{u}_0 = (\mathbf{u}^{\max} + \mathbf{u}^{\min})/2,\ \ \Delta\mathbf{U} = diag\left((u_i^{\max} - u_i^{\min})/2\right)$$

The summation is truncated to a finite number of N terms and the unknowns are the expansion coefficients $\mathbf{a}_i$. If we now expand the performance index $J(\mathbf{x}(\tau{=}1))$ in terms of the $\mathbf{a}_i$ constants, of the $\mathbf{u}(\tau)$ function can be written as:

$$J(u) = b_0 + \sum_{i=1}^{N} b_i a_i + \sum_{j=i}^{N}\sum_{i=1}^{N} b_{ij} a_i a_j + \sum_{k=j}^{N}\sum_{j=i}^{N}\sum_{i=1}^{N} b_{ijk} a_i a_j a_k + \ldots \qquad (4)$$

This will be called the Response Surface Model (RSM). For simplicity's sake, we have assumed in eq. (4) that there is only one decision function u(τ) and that each of the a_i constants is a scalar rather than a vector. The main model parameters are now the constants b_i, b_{ij}, b_{ijk} etc., relating the performance index J and the different choices of the decision variable u(τ). In the rare case that the knowledge-based process model is known *a priori*, the constants b can be explicitly calculated. Once the b constants are known, an optimization can be performed to calculate the optimal values of the parameters a_i (i=1, 2…n) that describe the estimate of the unknown optimal profile $u^*(\tau)$. *Of interest here is the circumstance in which* ***no model*** *for the batch process is available a priori.* In such a case, the novel approach introduced by the present paper consists of the following five steps:

a. Select a functional basis $\varphi_i(\tau)$ to parameterize the input function u(τ).
b. Design a set of time-varied experiments characterized by a properly selected set of constants a_i.
c. Perform the experiments.
d. Estimate the values of the b parameters in the RSM (eq. 4), using the values of J that correspond to each of the performed experiments.
e. Calculate the values of a_i that optimize J. Perform the optimal experiment and compare the results with the response surface model predictions.

The proposed approach is called Model-Free for two reasons. First, the RSM is a rather simple easy-to-develop interpolative model that contains no fundamental information about the process. Second, the process can be still substantially optimized my simply choosing the best of the initial dynamic experiments.

3.2. The Algorithmic Steps

We provide here some additional details of the four steps described before.

1. Define a dimensionless variable $w(\tau)$, referred as coded variable, that varies between -1 and +1 and which characterizes the time dependent process variable, or dynamic factor. For example, if the dynamic factor is the reactor temperature and it is allowed to vary between T_{max} and T_{min}, then the coded variable w(τ) is defined by:

$$w(\tau) = [2T(\tau) - (T_{\max} + T_{\min})]/(T_{\max} - T_{\min}) \qquad (5)$$

   In the case that we have more than one decision variable u(τ), we define the coded variable by

$$\mathbf{w}(\tau) = \Delta\mathbf{U}^{-1}\left(\mathbf{u}(\tau) - \mathbf{u}_0\right) \qquad (6)$$

2. Select an appropriate functional basis $\{\varphi_i(\tau)|\ i=1, 2,\ldots\}$ defined in the interval [0, 1]. These functions must be a linearly independent set that is complete and thus can serve as a functional basis. This functional basis could be either an orthogonal or a non-orthogonal one. The selection of this basis should be influenced by the expected character of the problem's solution in order to reduce the number of needed expansion terms and thus the number of experiments.

3. The unknown value of the dynamic factors $\mathbf{u}(\tau)$ that maximizes a certain performance index of the process $J(\mathbf{u})$ is denoted by, $\mathbf{u}^*(\tau)$:

$$J(\mathbf{u}^*) = \max_{\mathbf{u}(\tau)} J(\mathbf{u}) \qquad (7)$$

   The unknown vector function $\mathbf{u}^*(\tau)$ is expanded in terms of a linear combination of the basis functions $\varphi_i(\tau)$, given in eq. (3).

4. Substitute the optimization with respect to $\mathbf{u}(\tau)$ with an optimization with respect to the constants $\boldsymbol{a}_i$. For each component function $u_q(\tau)$ of $\mathbf{u}(t)$, the corresponding constants $a^q{}_i$ are called the sub-factors that characterize the unknown dynamic factor $u_q(\tau)$. The infinitely dimensional search for the optimal function $\mathbf{u}^*(\tau)$ is then substituted by a finite dimensional search of the pN constants a, where p is the dimensionality of $\mathbf{u}(\tau)$.

5. Design experiments motivated by the classical Design of Experiments (DoE) methodology for the selection of the appropriate values of the sub-factors $a^q{}_i$. Each set of values of the sub-factors correspond to a specific time-dependent function $u_j(\tau)$ or $w_j(\tau)$. However, one needs to take into account certain constraints that $u_j(\tau)$ or $w_j(\tau)$ will have to satisfy.
6. Develop an appropriate interpolating response surface model relating J to the values of the $a^q{}_i$ in the form of eq. (4). The unknown parameters of the model are the coefficients b_j, b_{ij}, b_{ijk} etc. and a linear regression algorithm can be used for their estimation. An analysis of variance (ANOVA) is performed to reveal which of the terms are the most significant based on the accuracy of the experimental measurements.
7. Calculate the optimal values of the $a^{q*}{}_i$ coefficients that optimize J. This is a constrained optimization task since each of the coefficients a_i is constrained by an upper and lower value (usually $-1 \le a_i \le +1$). The optimal values of the a_j determine the optimal function $u(\tau)$.

The methodology described above substitutes the unknown function $u(\tau)$ by its coefficients a_i. By selecting the appropriate values of the a_i, one designs dynamic experiments with several choices of the input function $u(\tau)$. Each of the experiments results in a value of the performance index J. The set of such values enables the calculation of the response surface model (RSM) of equation (4), which is used to optimize the process with respect to the decision variable(s) $u(\tau)$. The proposed methodology generalizes the classical design of experiments (DoE) (Montgomery, 2005) methodology with respect to dynamically varying processes. For this reason, the term Design of Dynamic Experiments (DoDE) was coined to describe it (Georgakis, 2008).

4. DESIGN OF THE DYNAMIC EXPERIMENTS

Here we present some example designs of the DoDE experiments. We select the (shifted) Legendre polynomial as the basis in the Hilbert space $\mathcal{L}_2(0,1)$. The first three Legendre polynomials are

$$\phi_1(\tau) = P_0(\tau) = 1, \quad \phi_2(\tau) = P_1(\tau) = 1 - 2\tau, \\ \phi_3(\tau) = P_2(\tau) = 1 - 6\tau + 6\tau^2, \ \ldots \tag{8}$$

We will use these orthogonal polynomials to define the dynamic experiments in all the examples discussed here.

4.1. Simple Example of the DoDE Design Approach

The simplest set of DoDE experiments is obtained by selecting the smallest value of N in eq. (3), equal to two. This implies that the dynamic profile of $u(\tau)$, or the coded variable $w(\tau)$, is a linear combination of the first two Legendre polynomials $P_0(\tau)$ and $P_1(\tau)$. This limits our consideration among constant or linear time dependencies. In deciding the values of the a_1 and a_2 sub-factors we can follow the classical DoE approach. If we do level-2 experiments for each sub-factor, we will design the following 2^2=4 experiments:

$$\begin{pmatrix} a_1 \\ a_2 \end{pmatrix} = \left[\begin{pmatrix} +1 \\ +1 \end{pmatrix}, \begin{pmatrix} +1 \\ -1 \end{pmatrix}, \begin{pmatrix} -1 \\ +1 \end{pmatrix}, \begin{pmatrix} -1 \\ -1 \end{pmatrix} \right] \tag{9}$$

Translating, for example, the second experiment to the corresponding dynamic coded variable $w(\tau)$ we have $w(\tau) = P_0(\tau) - P_1(\tau) = 2\tau$. We realize that this profile does not meet its constraints $-1 \le w(\tau) \le +1$. This is easily remedied by imposing the following constraints on the a_1 and a_2 constants: $-1 \le a_1 + a_2 \le +1$ and $-1 \le a_1 - a_2 \le +1$ reducing the values of a_1 and a_2, without changing their relative magnitude so the that the constraints on $w(\tau)$ are met. This leads to the following design of experiments

$$\begin{pmatrix} a_1 \\ a_2 \end{pmatrix} = \left[\begin{pmatrix} +0.5 \\ +0.5 \end{pmatrix}, \begin{pmatrix} +0.5 \\ -0.5 \end{pmatrix}, \begin{pmatrix} -0.5 \\ +0.5 \end{pmatrix}, \begin{pmatrix} -0.5 \\ -0.5 \end{pmatrix} \right] \tag{10}$$

In this case, the four time-variations in $w(\tau)$ are shown in Figure 1.

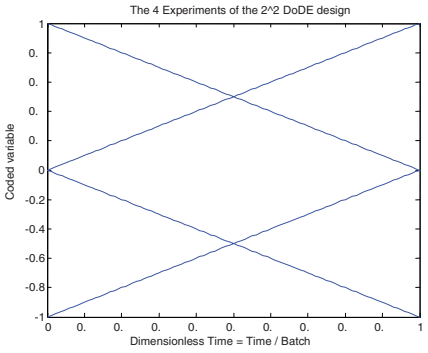


***Figure 1**: The 2^2=4 DoDE experiments*

4.2. Other DoDE Examples

For N=3, we are considering the first three Legendre polynomials, and if we consider only low and high values, we need to perform 2^3= 8 experiments. Here we are considering quadratic dependence on time along with the constant and linear sub-factors considered before. If we add cubic dependence by letting N=4 we need 2^4= 16 experiments and the use of the next Legendre polynomial: $P_3(\tau) = -20\tau^3 + 30\tau^2 - 12\tau + 1$. For N=5 we involve the first five shifted Legendre polynomials. This includes the four polynomials mentioned above along with the fifth one: $P_4(\tau) = 70\tau^4 - 140\tau^3 + 90\tau^2 - 20\tau + 1$. Here we need 2^5= 32 experiments. In the case we design level-3 full factorial experiments we need 3^2= 9 experiments for N=2, and 3^3= 27 experiments for N=3.

We should note that the way the dynamic experiments are designed involves two steps that are similar to the ones presented in section 4.1 for the simplest of the DoDE designs. In the first step, the sub-factors related to the values of the coefficients a_i are treated as independent from each other and are assigned initial values of (-1 , +1) in the level 2 designs and values (-1, 0, +1) in the lever 3 designs. In the second

Table 1: *Details of the Batch Reactor Optimization using the DoDE Methodology*

Dynamic Sub-factors	Levels	Case Number and Number of Experiments	Best Conversion from Initial 2^n or 3^n Experiments	Best Profile from the Initial 2^n or 3^n Experiments: $T(\tau)=308+15u(\tau)$ ^oK with $u(\tau)=a_1P_0+a_2P_1+a_3P_3+\ldots$	RSM-Optimum Conversion of A	Defining Parameters of the Calculated RSM-Optimum Profile $w(\tau)$
Level 2 FULL Factorial Designs						
1	2	DA1: $2^1=2$	62.32%	$a_1 = -1$	62.23%	$a_1 = -1$
2	2	DA2 : $2^2=4$	73.46%	$(a_1, a_2) = (-0.5, 0.5)$	76.57%	$(a_1, a_2) = (0, 1)$
3	2	DA3: $2^3=8$	74.61%	$(a_1, a_2, a_3) = (-0.3, 0.3, -0.3)$	77.77%	$(a_1, a_2, a_3) = (0, 1, 0)$
4	2	DA4: $2^4=16$	74.82%	$(a_1, a_2, a_3, a_4) = (-0.3, 0.3, -0.3, 0.3)$	78.10%	$(a_1, a_2, a_3, a_4) = (0, 1, 0, -0.04)$
5	2	DA5: $2^5=32$	74.43%	$(a_1, a_2, a_3, a_4, a_5) = (-0.3, 0.3, -0.3, 0.3, .3)$	78.43%	$(a_1, a_2, a_3, a_4, a_5) = (0, 0.9, 0, -0.08, -0.2)$
Level 3 FULL Factorial Designs						
1	3	DA6 $3^1=3$	73.91%	$a_1 = 0$	73.92%	$a_1 = -0.03$
2	3	DA7 $3^2=9$	77.35%	$(a_1, a_2) = (0, 1)$	77.57%	$(a_1, a_2) = (0.1, 0.9)$
3	3	DA8 $3^3=27$	77.35%	$(a_1, a_2, a_3) = (0, 1, 0)$	77.66%	$(a_1, a_2, a_3) = (0.05, 0.9, 0.06)$

step, all of the a_i values related to a single experiment are scaled up or, in most cases, down by a common factor, so that the coded dynamic variable $w(\tau)$ attains values that are inside the [-1, +1] interval. Making the maximum (or minimum) of each profile touch the maximum (or minimum) values of $w(\tau)$ also ensures that the set of DoDE experiments covers all areas on the [-1, +1] x [0, 1] rectangle.

5. BATCH REACTOR WITH REVERSIBLE REACTION

Here we consider the optimization of the operation of a batch reactor in which a reversible reaction between reactant A and product B takes place with the following characteristics:

$$A \underset{k_2}{\overset{k_1}{\rightleftharpoons}} B; \quad r = k_1 C_A - k_2 C_B \quad with$$

$$k_1 = k_{10}\exp(-E_1/RT)\ [1/hr]; \quad k_2 = k_{20}\exp(-E_2/RT)\ [1/hr]$$

$$k_{10}=1.32\text{x}10^7; \quad k_{20}=5.24\text{x}10^{13}; \quad E_1 = 10{,}000; \quad E_2 = 20{,}000$$

We select the activation energy of the reverse reaction to be larger than that for the forward reaction. This leads to the expectation that the optimum temperature profile is a decreasing one (Rippin, 1983). One needs to note here, that for the development of the fundamental model above, we need to *assume* that the first order kinetic rate is correct. We then perform *at least* 4 experiments to estimate the values of k_{10}, k_{20}, E_1, and E_2.

With such a model at hand, one can optimize the reactor temperature profile to maximize the conversion of reactant A. This is achieved by converting the ODEs into algebraic equations via Radau collocation on finite elements (Biegler, 2007). The reactor temperature is constrained between 20 ^{0}C and 50 ^{0}C and the optimization is achieved by use of the IPOPT algorithm (Wächter & Biegler, 2006).

The optimum profile calculated is constant at the upper temperature constraint for almost 0.4 hrs and then decreases to the minimum constraint at the end of the batch. We select here to fix the batch time to 2.5 hrs and the maximum conversion of the reactant A is calculated to be 77.68%. In Table 1, the results of the different DoDE experiments are presented. They involve up to 5 dynamic sub-factors and include level-2 (low-high) and level-3 (low-medium-high) experiments. In the fourth column the best conversion value of the initial runs is given. In the second to last column, the expected best batch performance, as calculated by the optimization of the response surface model (RSM), is given. In the last column the characteristics of this RSM-optimal profile is given in terms of the coded variable: $w(\tau) = a_1P_0(\tau) + a_2P_1(\tau) + \ldots$ The temperature profile can then be calculated by eq. 5. We observe that in the case denoted as DA2 in Table 1 *only* four experiments described in Figure 1 yield an RSM-optimum with a conversion of 76.57% which is just 1.43% away from the true optimum of 77.68%. We also observe that a larger number of experiments, such as those of cases DA4, DA5, DA7 and DA8, predict a higher conversion, even closer to the true optimum. However, as the number of experiments performed increases, the changes in the predicted optimum conversion becomes smaller and smaller per additional experiment, implying that the true optimum has been reached.

6. PENICILLIN FERMENTATION

Here we simulate the penicillin fermentation model of Bajpai and Reuss (Bajpai & Reuss, 1980) which has been the center of attention in several model-based optimizations (Riascos & Pinto, 2004). The model used to simulate the experiment consists of the equations in the Appendix. To focus on the main idea of calculating the optimum *time-varying* profile, we fix the batch time to t_b=130 hrs and the growth phase of the biomass to t_f=30 hrs. Here we want to demonstrate the application of the DoDE approach to this challenging optimization problem to demonstrate its power in optimizing complex processes. For this reason, we are *not* designing experiments that vary the t_f and t_b values, since they are not time-varying decision variables or factors.

We choose to design 16 experiments with $4=2^2$ variations for the substrate flow in growth phase $0 \le t \le t_f$ *(or* $0 \le \tau \le \gamma$; $\gamma = t_f/t_b$*)*

and $4=2^2$ additional variations for the production phase $t_f \le t \le t_b$ *(or* $\gamma \le \tau \le 1$*)*, the profiles are depicted in Figure 2. In the growth phase, the average value of the substrate flow is 30 gr/hr, with an allowed change up and down of 20 *gr/hr*. In the production phase, the average flow considered is 7 *gr/hr* and it varies by 3 *gr/hr*, up or down. Each of the four feeding profiles in the first phase is combined with each of the four feeding profiles of the second phase.

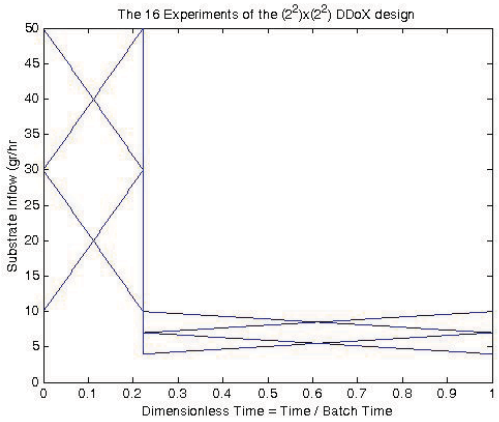


Figure 2: *The $2^2x2^2=14$ DoDE Experiments for Penicillin Fermentation (Total Volume Unconstrained)*

Some of these designs result in an increase in the bioreactor volume by more than 4 lt. from the initial value of 7*lt*. For example, one such profile is the one identified with the following sub-factor coefficients (a_{11}, a_{12}, a_{21}, a_{22}) = (0.5, 0.5, 0.5, 0.5). To meet the final volume constraint of 11*lt*, we impose the following total volume constraint:

$$(u_{1m} + \Delta u_1 * a_{11})t_f + (u_{2m} + \Delta u_2 * a_{21})(t_b - t_f) \le (V_T - V_0)s_f$$

with

$$u_{1m} = 30gr/hr,\ u_{2m} = 7gr/hr,\ \Delta u_1 = 20gr/hr,\ \Delta u_2 = 3gr/hr$$

$$t_b = 130hr,\ t_f = 30hr,\ V_0 = 7lt \text{ and } V_T = 11lt$$

This profile is modified to the (0.44, 0.5, 0.44, 0.5) one (DB16). Three additional profiles need such modification. The resulting time dependencies on the overall feeding profiles are defined in Table 2. The resulting final bioreactor volume and total amount of penicillin produced (i.e. the performance index *J*) are also given in Table 2.

The time evolution of the simulated process during experiments DB9 is given in Figure 3. Using all the data of Table 2, a response surface model is estimated and constrained optimization, $V(t_b)<11$, yields an optimum of the penicillin process with the production of $J=102.30$ grams of product. The calculated optimum feeding profile is characterized by the following values of the 2+2 sub-factor coefficients: (a_{11}, a_{12}, a_{21}, a_{22}) = (0.19, 0, 0.95, 1.0). Simulation of this operation yields 104.17 grams of product, a bit more than predicted.

Here we used a level 2 experimental design which necessitates that the response surface model has only linear and interaction terms. No quadratic terms are allowed. A more accurate response surface model can be constructed if one uses a level 3 DoDE design. In such a case, the response surface model includes quadratic terms.

Table 2: *Definition and Performance Index of the 16 Volume-Constrained Penicillin Experiments*

Run Label	Growth Phase		Production Phase		$V(t_f)$ [lt]	$J=V(t_f)p(t_f)$ [gr]
	a_{11}	a_{12}	a_{21}	a_{22}		
DB1	-0.5	-0.5	-0.5	-0.5	9.3	+58.97
DB2	-0.5	+0.5	-0.5	-0.5	9.3	+59.06
DB3	+0.5	-0.5	-0.5	-0.5	10.5	+57.93
DB4	+0.5	+0.5	-0.5	-0.5	10.5	+57.93
DB5	-0.5	-0.5	-0.5	+0.5	9.3	+54.05
DB6	-0.5	+0.5	-0.5	+0.5	9.3	+54.15
DB7	+0.5	-0.5	-0.5	+0.5	10.5	+51.82
DB8	+0.5	+0.5	-0.5	+0.5	10.5	+51.83
DB9	-0.5	-0.5	+0.5	-0.5	9.9	+84.41
DB10	-0.5	+0.5	+0.5	-0.5	9.9	+84.46
DB11	**+0.44**	**-0.5**	**+0.44**	**-0.5**	**11.0**	**+86.08**
DB12	**+0.44**	**+0.5**	**+0.44**	**-0.5**	**11.0**	**+86.08**
DB13	-0.5	-0.5	+0.5	+0.5	9.9	+81.33
DB14	-0.5	+0.5	+0.5	+0.5	9.9	+81.40
DB15	+0.44	-0.5	+0.44	+0.5	11.0	+80.01
DB16	+0.44	+0.5	+0.44	+0.5	11.0	+80.01

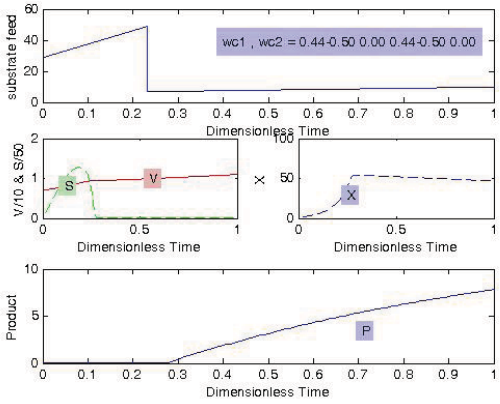


Figure 3: *The time evolution of the state and input variables of the $2^2x2^2=16$ DoDE experiments for penicillin fermentation (total volume constrained)*

We observe that the RSM-optimal run yields a performance index that is 20.82% better than any of the initial 16 experiments. This is very significant but this is not the major result of this investigation. The major result is that the process is significantly optimized with just the 16 initial

systematically designed DoDE experiments. This is a much smaller effort than what is needed to develop a fundamental model describing the process, necessary for the classical approach in process optimization.

7. CONCLUSIONS

We presented a new approach to optimize batch processes with respect to one or more time-varying decision variables. The method, called Design of Dynamic Experiments (DoDE), defines a set of experiments in which time-varying patterns of the decision variable is used. A response surface model, built from the performance index values of each experiment, is used to optimize the process. Two examples, a batch reaction and penicillin fermentation, are used to demonstrate the powerful characteristics of the new methodology. Due to space limitations, we have not presented the related ANOVA analysis. The effect of measurement error (1%-5%) was investigated and it has been convincingly shown that its effect on the process optimization is not at all detrimental to the proposed approach.

8. REFERENCES

Bajpai, R. K., & Reuss, M. (1980). A Mechanistic Model for Penicillin Production. *Journal of Chemical Technology and Biotechnology, 30*(6), 332-344.

Biegler, L. T. (1984). Solution of Dynamic Optimization Problems by Successive Quadratic Programming and Orthogonal Collocation. *Computers & Chemical Engineering, 8*(3-4), 243-247.

Biegler, L. T. (2007). An overview of simultaneous strategies for dynamic optimization. *Chemical Engineering and Processing, 46*(11), 1043-1053.

Box, G. E. P., & Draper, N. R. (2007). *Response Surfaces, Mixtures, and Ridge Analysis*. Hoboken, NJ: Wiley.

Cabassud, M., Cognet, P., Garcia, V., Le Lann, M. V., Casamatta, G., & Rigal, L. (2005). Modeling and optimization of lactic acid synthesis by the alkaline degradation of fructose in a batch reactor. *Chemical Engineering Communications, 192*(6), 758-786.

Fotopoulos, J., Georgakis, C., & Stenger, H. G. (1996, May 05-08). *Effect of process-model mismatch on the optimization of the catalytic epoxidation of oleic acid using tendency models.* Paper presented at the 14th International Symposium on Chemical Reaction Engineering - From Fundamentals to Commercial Plants and Products, Brugge, Belgium.

Fotopoulos, J., Georgakis, C., & Stenger, H. G. (1998). Use of tendency models and their uncertainty in the design of state estimators for batch reactors. *Chemical Engineering and Processing, 37*(6), 545-558.

François, G., Srinivasan, B., & Bonvin, D. (2005). Use of measurements for enforcing the necessary conditions of optimality in the presence of constraints and uncertainty. *Journal of Process Control, 15*(6), 701-712.

Georgakis, C. (2008). Dynamic Design of Experiments for the Modeling and Optimization of Batch Process.

Kameswaran, S., & Biegler, L. T. (2006, Jan 08-13). *Simultaneous dynamic optimization strategies: Recent advances and challenges.* Paper presented at the 7th International Conference on Chemical Process Control (CPC 7), Lake Louise, CANADA.

Kameswaran, S., & Biegler, L. T. (2008). Convergence rates for direct transcription of optimal control problems using collocation at Radau points. *Computational Optimization and Applications, 41*(1), 81-126.

Martinez, E. C. (2005, May 29-Jun 01). *Model discrimination and selection in evolutionary optimization of batch processes with tendency models.* Paper presented at the 15th European Symposium on Computer Aided Process Engineering (ESCAPE-15), Barcelona, SPAIN.

Montgomery, D. C. (2005). *Design and Analysis of Experiments* New York: Wiley.

Riascos, C. A. M., & Pinto, J. M. (2004). Optimal control of bioreactors: a simultaneous approach for complex systems. *Chemical Engineering Journal, 99*(1), 23-34.

Rippin, D. W. T. (1983). Simulation of a Single- and Multiproduct Batch Chemical Plants for Optimal Design and Operation *Computers & Chemical Engineering, 7*(3), 137-156.

van den Bos, A. (2007). *Parameter Estimation for Scientists and Engineers* John Wiley & Sons, Inc. .

Wächter, A., & Biegler, L. T. (2006). On the implementation of an interior-point filter line-search algorithm for large-scale nonlinear programming. *Mathematical Programming 106* (1), 25.

9. ACKNOWLEDGMENTS

The author wishes to acknowledge the assistance of Victor Zavala and Larry Biegler in the selection of the AMPL and IPOPT software environment for calculation of the Model-based Optimum of the first example process. Alex Marvin, a ChBE senior undergraduate student at Tufts University calculated the simulation data used in Table 1.

10. APPENDIX

In this Appendix we present the penicillin model used:

$$x_1 = V: \quad \frac{dV}{dt} = u / s_f$$

$$x_2 = x: \quad \frac{dx}{dt} = \mu x - \frac{x}{s_f V} \qquad \mu = \mu_{\max} \frac{s}{k_x x + s}$$

$$x_3 = s: \quad \frac{ds}{dt} = -\frac{\mu x}{Y_{x/s}} - \rho \frac{x}{Y_{p/s}} - \frac{m_s s}{k_m + s} x + \left(1 - \frac{s}{s_f}\right) \frac{u}{V}$$

$$x_4 = p: \quad \frac{dp}{dt} = \rho x - kd - \frac{p}{s_f V} \mathrm{u} \qquad \rho = \rho \max \left(\frac{s}{k_p + s + s^2 / k_{in}} \right)$$

The model parameters used are the ones reported in (Riascos & Pinto, 2004)

Controller Tuning

Oral Session

An Internal Model Control Approach to Mid-Ranging Control

Sandira Gayadeen*, William Heath*

*Control Systems Centre, School of Electrical and Electronic Engineering, The University of Manchester, PO Box 88, Sackville Street, Manchester M60 1QD UK (e-mail:sandira_vg@hotmail.com, William.Heath@manchester.ac.uk)

Existing tuning rules for mid-ranging control can be improved. In this paper a novel strategy for mid-ranging control based on Internal Model Control (IMC) principles is presented. The design reformulates mid-ranging control specifications in terms of classical bandwidth and sensitivity requirements. The performance of this design is demonstrated through simulation studies. The overall benefits of the IMC design are that it provides transparent and flexible tuning, and that it offers a natural framework for anti-windup. Both classical IMC and modified IMC structures are considered for anti-windup. Their performance during saturation is demonstrated through simulation studies, where minimal degradation is observed.

Keywords: control system analysis, control system design, controller, constraints, saturation

1. INTRODUCTION

The term mid-ranging control typically refers to the class of control problems where two control inputs i.e. actuators are manipulated to control one output. Furthermore there is the condition that one input should return to its midpoint or some setpoint. The inputs usually differ in their dynamic effect on the output and in the relative cost of manipulating each one with the faster input normally being more costly to use than the slower input (Henson et al., 1995). Therefore mid-ranging control schemes seek to manipulate both inputs upon an upset but then gradually reset or mid-range the fast input to its desired setpoint (Allison and Ogawa, 2003).

Mid-ranging control is commonly implemented using the architecture shown in Fig. 1 where u_1 is the fast input and u_2 is the slow input. This structure is referred to as Valve Position Control (VPC) and C_1 is usually chosen as a PI controller and C_2 as an I-only controller. The VPC method for mid-ranging has been found to be sub-optimal (Allison and Isaksson, 1998, Allison and Ogawa, 2003). As such, improvements to the approach of mid-ranging control problems are suggested by many authors. Model predictive control (MPC) has also been suggested by Allison and Isaksson (1998) as an advantageous approach to mid-ranging given that it is inherently a multi-variable control problem. Henson et al. (1995) also propose MPC as well as a Direct Synthesis approach for the design of habituating controllers. The habituating control described by Henson et al. (1995) is essentially a mid-ranging control problem. Allison and Ogawa (2003) put forward a Modified Valve Position Control (MVPC) scheme which combines the simplicity of conventional VPC with the systematic tuning of Direct Synthesis. Allison and Ogawa (2003) compare the performance of MVPC with that of both conventional VPC and Direct Synthesis.

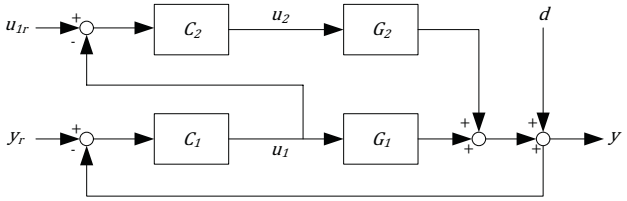


Fig. 1 Block diagram of VPC strategy (Allison and Ogawa, 2003)

For many applications MVPC works fine and has the advantage that it can be implemented using the standard VPC structure in Fig. 1 with PID control blocks. However, MVPC is not optimal; Henson et al. (1995) show that better performance (and implicitly better robustness) can be obtained by using a more general structure which includes both feedforward and feedback elements. This is acknowledged by Allison and Ogawa (2003). The Direct Synthesis design, unlike MVPC, also allows enhanced performance such as decoupling between u_{1_r} and y. MVPC does not achieve this decoupled response though u_1 tracks changes in u_{1_r} correctly.

The mid-ranging design proposed by Henson et al. (1995) uses both feedback and pre-filters. The design criteria are focused on:

- obtaining a desired response from y_r to y
- obtaining a desired response from u_{1_r} to u_1
- obtaining a decoupled response from u_{1_r} to y.

In this paper, a similar general structure is utilised where the decoupling can be achieved through the use of pre-filters.

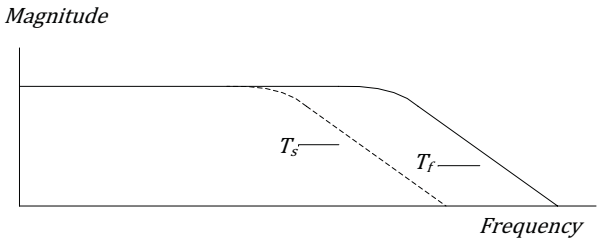


Fig. 2 Desired frequency response of complementary sensitivities

The mid-ranging design proposed in this paper focuses on the respective disturbance responses of y, u_1 and u_2 and exploits both the structure and tuning methodology of internal model control (IMC). This makes the design trade-offs transparent.

Allison and Ogawa (2003) do not discuss anti-windup for MVPC. However Haugwitz et al. (2005) have shown that for some applications an additional feedback block can significantly improve the anti-windup performance of MVPC. The IMC structure provides a natural framework for both anti-windup (discussed in this paper) and robustness analysis (see Morari and Zafiriou, 1989, for the general case).

In Sections 2 to 5 the IMC tuning method for mid-ranging is presented followed by simulation studies in Section 6 that demonstrate the performance of IMC compared to Direct Synthesis. Anti-windup in IMC mid-ranging is discussed in Section 7 with an example that shows how the classical IMC structure presented in the previous sections and a modified IMC structure perform during saturation of the inputs.

2. MID-RANGING CONTROL OBJECTIVES

The plant model is (see Fig. 1),

$$y = G_1 u_1 + G_2 u_2 + d$$

where u_1 is the fast input and u_2 is the slow input. The objective is to use the both inputs to control y and mid-range u_1 i.e. to return u_1 to its setpoint, u_{1_r}.

The transfer function between y_r and y can be defined as the fast complementary sensitivity, T_f. The response from y_r to y when u_1 is set to zero can be defined as the slow complementary sensitivity, T_s (corresponding to the control action with the slow actuator alone). These are chosen to produce desired responses to setpoint changes such that the frequency response looks like Fig. 2. The proposed IMC mid-ranging design is to specify not only T_f but also T_s. With two degrees of freedom, the rest follows as illustrated in Sections 3 and 4.

3. IMC STRUCTURE FOR MID-RANGING

Firstly the general IMC structure shown in Fig. 3 is considered. Assuming that the model is perfect ($G = \tilde{G}$), y and u can be derived as:

$$y = Gu + d \tag{1}$$

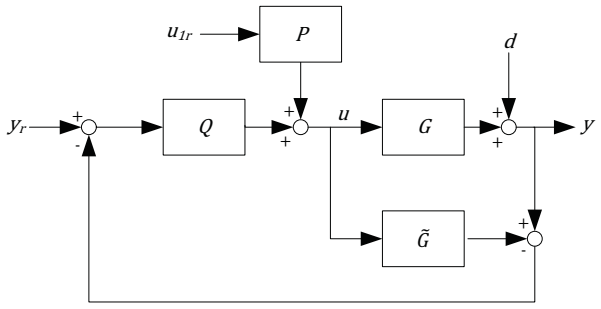


Fig. 3 IMC mid-ranging structure

$$u = Q(y_r - y + Gu) + Pu_{1_r} \tag{2}$$

Equations (1) and (2) can expressed as:

$$u = Qy_r - Qd + Pu_{1_r} \tag{3}$$

$$y = GQy_r + (I - GQ)d + GPu_{1_r} \tag{4}$$

For classical feedback control,

$$u = C(y_r - y) + \hat{P}u_{1_r}$$

where C is the equivalent feedback controller and $\hat{P}$ is the equivalent pre-filter.

C can be found by expressing (2) as:

$$u = (I - QG)^{-1}Q(y_r - y) + (I - QG)^{-1}Pu_{1_r} \tag{5}$$

This gives:

$$C = (I - QG)^{-1}Q$$
$$= Q(I - GQ)^{-1} \tag{6}$$

For mid-ranging the following are defined as:

$$u = \begin{bmatrix} u_1 \\ u_2 \end{bmatrix}, G = [G_1 \quad G_2], Q = \begin{bmatrix} Q_1 \\ Q_2 \end{bmatrix} \text{ and } P = \begin{bmatrix} P_1 \\ P_2 \end{bmatrix}$$

4. IMC MID-RANGING DESIGN

From (4), T_f and T_s can be expressed in terms of Q_1 and Q_2. This gives the conditions for the controller design:

Controller Design Conditions:

$$T_f = GQ = G_1Q_1 + G_2Q_2$$

$$T_s = G_2Q_2$$

The sensitivity, S_f is defined as $S_f = 1 - T_f$ such that (4) can be expressed as,

$$y = T_f y_r + S_f d + GPu_{1_r}$$

To achieve the control objectives,

$$T_f|_{ss} = G_1Q_1 + G_2Q_2|_{ss} = 1$$

$$Q_1|_{ss} = 0$$

This means that $T_s|_{ss} = G_2Q_2|_{ss} = 1$.

The decoupling between u_{1_r} and y is achieved through the use of pre-filters. From (4), to obtain a decoupled response between u_{1_r} and y, the following condition must be satisfied:

Pre-filters Design Conditions:

$GP = G_1P_1 + G_2P_2 = 0$

The design algorithm is then as follows:

1. Choose T_s
2. Q_2 is then designed such that $Q_2 = \frac{T_s}{G_2}$
3. Choose T_f
4. Q_1 is then designed such that $Q_1 = \frac{(T_f - T_s)}{G_1}$
5. To design pre-filters, $GP = 0$ must be satisfied. There are different ways to achieve this:
   a. Let $P_1 = 1$ and $P_2 = -\ G_1/G_2$
   b. Let $P_1 = \ G_2/G_2|_{ss}$ and $P_2 = -\ G_1/G_2|_{ss}$
   c. Let $G_2 = \ {G_2}^- {G_2}^+$ where ${G_2}^+$ is minimum phase and ${G_2}^-$ is non-minimum phase, then let $P_1 = H{G_2}^-$ where $H{G_2}^-|_{ss} = 1$ and $P_2 = -H\ G_1/{G_2}^+$

G_1 and G_2 can be either minimum phase or non-minimum phase. This design above can be extended for non-minimum phase systems as follows:

1. Choose T_s such that $T_s = {T_s}^-\tilde{T}_s$ where ${T_s}^-$ includes the non-minimum phase components (including delays) of both G_1 and G_2 i.e. so that T_s/G_1 and T_s/G_2 are both casual and stable.
2. Q_2 is then designed such that $Q_2 = \frac{T_s}{G_2}$
3. Choose $T_f = {T_f}^-\tilde{T}_f$ where ${T_f}^-$ includes the non-minimum phase component (including delay) of G_1 i.e. so that T_f/G_1 is casual and stable.
4. Q_1 is then chosen as $Q_1 = \frac{(T_f - T_s)}{G_1}$.
5. P_1 and P_2 are chosen as before.

The IMC approach concentrates on the disturbance responses i.e. the third column of the sensitivities matrices (17) and (28) in Henson et al. (1995) where nine sensitivities are discussed. Equation (7) shows the sensitivities for the IMC design.

$$\begin{bmatrix} y \\ u_1 \\ u_2 \end{bmatrix} = \begin{bmatrix} GQ & GP & (I - GQ) \\ Q_1 & P_1 & -Q_1 \\ Q_2 & P_2 & -Q_2 \end{bmatrix} \begin{bmatrix} y_r \\ u_{1_r} \\ d \end{bmatrix} \quad (7)$$

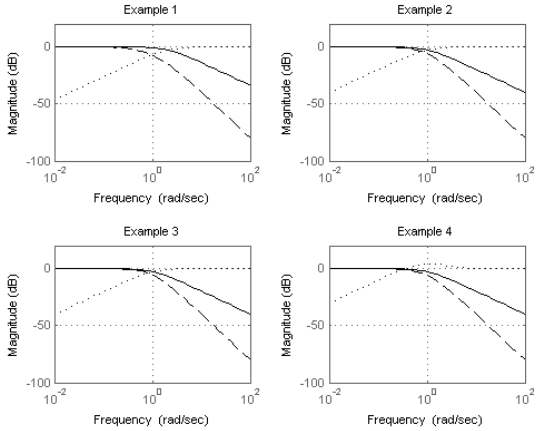


Fig. 4 Complementary sensitivities T_f (solid) and T_s (dashed) and sensitivity, S_f (dotted) for all examples.

From a classical point of view, the disturbance responses correspond to the sensitivity S_f and two control sensitivities, Q_1 and Q_2. For the mid-ranging problem, it is required that the slow actuator should have a control sensitivity that is low bandwidth only; meanwhile the fast actuator should have a control sensitivity that is mid-frequency only and goes to zero at steady state, giving mid-ranging. IMC is advantageous because it gives the control sensitivities of the fast and slow actuators directly as Q_1 and Q_2.

5. IMPLEMENTATION IN VPC STRUCTURE

The IMC mid-ranging design described in Section 4 can be extended for implementation using the VPC structure. This structure is similar to Fig. 1 but includes the pre-filters P_1 and P_2. Without the pre-filters this design does not achieve the decoupled response between u_{1_r} and y; it can only track u_1 to $u_1 = u_{1_r}$.

For the VPC structure in Fig. 1,

$$\begin{bmatrix} u_1 \\ u_2 \end{bmatrix} = \begin{bmatrix} C_1 \\ -C_2C_1 \end{bmatrix} (y_r - y)$$

From (6),

$$\begin{bmatrix} C_1 \\ -C_2C_1 \end{bmatrix} = \begin{bmatrix} Q_1 \\ Q_2 \end{bmatrix} \left[I - \begin{bmatrix} G_1 & G_2 \end{bmatrix} \begin{bmatrix} Q_1 \\ Q_2 \end{bmatrix} \right]^{-1}$$

Therefore,

$$\begin{bmatrix} C_1 \\ -C_2C_1 \end{bmatrix} = \begin{bmatrix} Q_1/(I - G_1Q_1 - G_2Q_2) \\ Q_2/(I - G_1Q_1 - G_2Q_2) \end{bmatrix}$$

This gives,

$$C_1 = \frac{Q_1}{(I - G_1Q_1 - G_2Q_2)}$$

$$C_2 = -\frac{Q_2}{Q_1}$$

6. SIMULATION STUDIES

The performance of the IMC design in the previous sections is demonstrated via simulation of four examples. The examples are all linear, stable systems. Examples 1, 2 and 3 are taken from Henson et al., (1995) to compare the performance of IMC controllers to the Direct Synthesis controllers. Therefore T_f and T_s for examples 1 to 3 are chosen to correspond to g_{y_d} and g_{u_d} given in Henson et al. (1995). In example 1, both G_1 and G_2 are minimum phase systems. For examples 2 and 3, G_1 is minimum phase whereas G_2 has a right half plane zero and a time delay respectively. Example 4 is an original example where both G_1 and G_2 are non-minimum phase. Additional first order setpoint filters are included in both control schemes according to Henson et al. (1995).

The process model transfer functions for the examples are given below.

Example 1: $y = \frac{1}{2s+1}u_1(s) + \frac{1}{5s+1}u_2(s) + \frac{1}{s+1}d(s)$

Example 2: $y = \frac{1}{2s+1}u_1(s) + \frac{-2s+1}{(2s+1)^2}u_2(s) + \frac{1}{s+1}d(s)$

Example 3: $y = \frac{1}{2s+1}u_1(s) + \frac{e^{-2s}}{2s+1}u_2(s) + \frac{1}{s+1}d(s)$

Example 4: $y = \frac{-s+1}{(2s+1)^2}u_1(s) + \frac{-2s+1}{(2s+1)^2}u_2(s) + \frac{1}{s+1}d(s)$

T_f and T_s are as follows:

Example 1: $T_f = \frac{1}{0.5s+1}$ and $T_s = \frac{2}{(2s+1)(s+2)}$

Example 2: $T_f = \frac{1}{s+1}$ and $T_s = \frac{-2s+1}{(2s+1)(s+1)^2}$

Example 3: $T_f = \frac{1}{s+1}$ and $T_s = \frac{e^{-2s}}{(s+1)^2}$

Example 4: $T_f = \frac{-s+1}{(s+1)^2}$ and $T_s = \frac{(-s+1)(-2s+1)}{(2s+1)(s+1)^3}$

The frequency response of sensitivity, S_f and the complementary sensitivities, T_f and T_s for the examples are shown in Fig. 4. T_f is chosen to determine the closed-loop bandwidth and T_s to determine the relative work done by u_1 and u_2.

Figs. 5 to 8 show setpoint and disturbance responses for the IMC controllers (solid line) and Direct Synthesis controllers, both parallel (dashed line) and series (dotted line).

It can be seen from Figs. 5 to 7 that identical output responses are obtained for Direct Synthesis and IMC schemes. This is expected since both pairs of controllers are tuned with the same time constants. The Direct Synthesis parallel architecture and IMC structure are identical when (6) is satisfied where C_1 and C_2 are the habituating controllers, g_{c11} and g_{c21} from Henson et al. (1995).

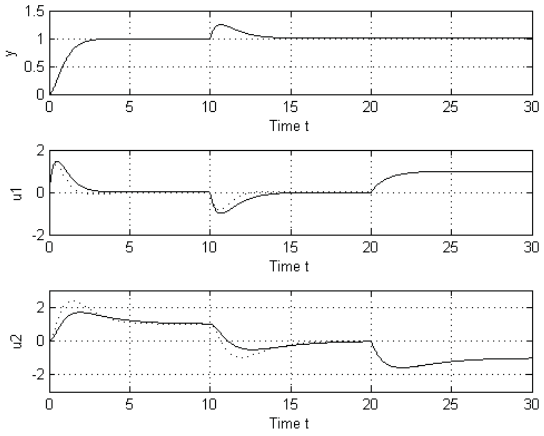


Fig. 5 Example 1: IMC (solid), Direct Synthesis parallel (dashed) and series (dotted) responses to unit changes in y_r at t=0, d at t=10 and u_{1_r} at t=20.

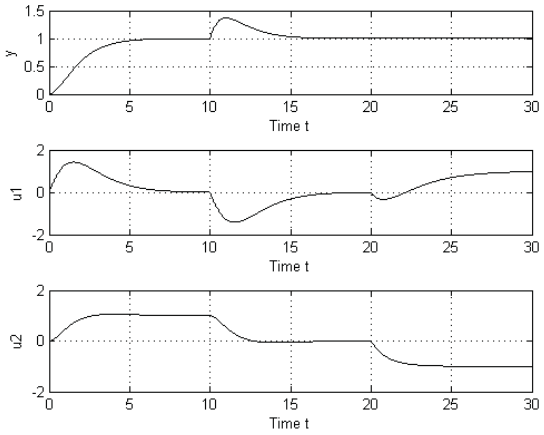


Fig. 6 Example 2: IMC (solid), Direct Synthesis parallel (dashed) and series (dotted) responses to unit changes in y_r at t=0, d at t=10 and u_{1_r} at t=20.

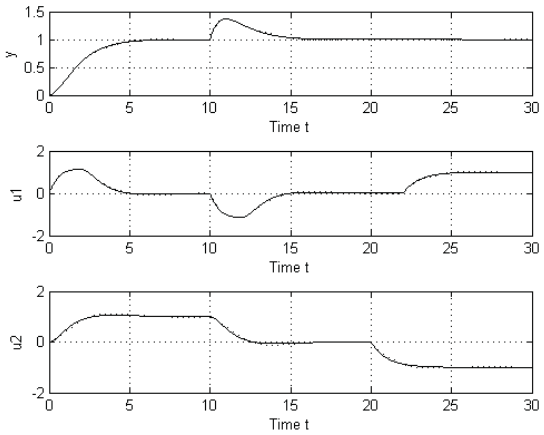


Fig. 7 Example 3: IMC (solid), Direct Synthesis parallel (dashed) and series (dotted) responses to unit changes in y_r at t=0, d at t=10 and u_{1_r} at t=20.

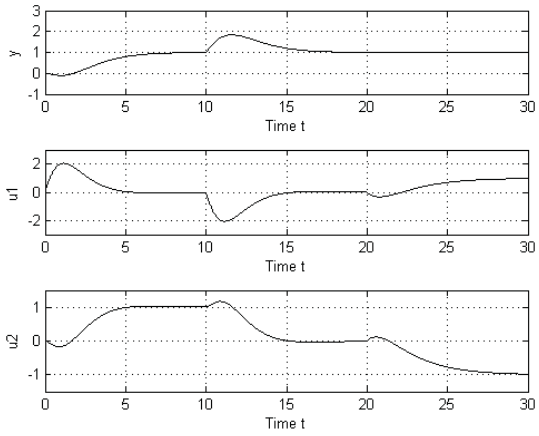


Fig. 8 Example 4: IMC (solid) response to unit changes in y_r at t=0, d at t=10 and u_{1_r} at t=20.

Direct Synthesis implemented in the series architecture responds differently from the IMC and the parallel architecture in example 1 because the relationship given by (11) in Henson et al. (1995) is not satisfied (because of the choice of $g_{c_{21}}$ in the design). For all examples, both methods produce completely decoupled response between u_{1_r} and y. However the IMC design gives more flexibility since H in Section 4 can be adjusted to tune the response to allow faster setpoint tracking of u_{1_r}.

From (7), Q_1 and Q_2 are directly related to the output setpoint and disturbance responses. Therefore it is simple to adjust these controllers because in the absence of model uncertainty, closed loop stability is automatically guaranteed as long as Q_1 and Q_2 are stable (Prett and Garcia, 1988). Henson et al. (1995) state that the controller $g_{c_{21}}$ can also be used to tune the responses of the two inputs to changes in y_r and d. The effect on closed loop performance of adjusting this tuning parameters is however not obvious. Neither is it obvious for what parameter values the closed loop system is stable.

7. ANTI-WINDUP IN MID-RANGING CONTROL

Haugwitz et al. (2005) propose anti-windup schemes for MVPC when u_1 saturates. Guidelines are given on how to tune mid-ranging controllers to maintain the same control action of u_2 in the saturated case as in the unsaturated case. Furthermore a modified anti-windup scheme is presented that achieves increased control action in u_2 to further reduce performance degradation. However, if the MVPC structure is to be modified, then significantly improved performance can be achieved for the saturated case by using the IMC approach in this paper.

In this section, the performance of the classical IMC structure used for mid-ranging (see Fig. 3) is considered when the inputs u_1 and u_2 saturate. This IMC structure works for most cases but as demonstrated by Zheng et al. (1994), sometimes IMC requires a modified structure. The modified IMC structure presented by Zheng et al. (1994) can be utilised for

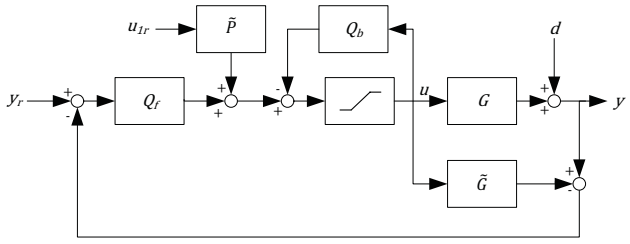


Fig. 9 Modified IMC mid-ranging structure for anti-windup

the proposed IMC mid-ranging design as shown in Fig. 9. The performance during saturation of this modified IMC structure (Fig. 9) is also considered in this section.

7.1 Anti-windup in IMC

As with the controller design, firstly the general case is considered. For the unsaturated case in Fig. 9,

$$u = (1 + Q_b)^{-1} Q_f (y_r - y + Gu) + (1 + Q_b)^{-1} \tilde{P} u_{1_r}$$

From (2),

$$u = Q(y_r - y + Gu) + P u_{1_r}$$

Therefore it is desired that,

$$Q = (1 + Q_b)^{-1} Q_f \tag{7}$$

$$P = (1 + Q_b)^{-1} \tilde{P}$$

For mid-ranging, u, G, Q and P are defined as in Section 3. Additionally, $\tilde{P}$, Q_f and Q_b are as follows:

$$\tilde{P} = \begin{bmatrix} \tilde{P}_1 \\ \tilde{P}_2 \end{bmatrix}, \; Q_f = \begin{bmatrix} Q_{f_1} \\ Q_{f_2} \end{bmatrix} \text{ and } Q_b = \begin{bmatrix} Q_{b_1} & 0 \\ 0 & Q_{b_2} \end{bmatrix}.$$

Extending the general case before for mid-ranging gives:

$$\begin{bmatrix} u_1 \\ u_2 \end{bmatrix} = \left(I + \begin{bmatrix} Q_{b_1} & 0 \\ 0 & Q_{b_2} \end{bmatrix} \right)^{-1} \begin{bmatrix} \bar{u}_1 \\ \bar{u}_2 \end{bmatrix}$$

$$\text{where } \begin{bmatrix} \bar{u}_1 \\ \bar{u}_2 \end{bmatrix} = \begin{bmatrix} \tilde{P}_1 \\ \tilde{P}_2 \end{bmatrix} u_{1_r} + \begin{bmatrix} Q_{f_1} \\ Q_{f_2} \end{bmatrix} \left(y_r - y + [G_1 \quad G_2] \begin{bmatrix} u_1 \\ u_2 \end{bmatrix} \right)$$

Therefore,

$$\begin{bmatrix} u_1 \\ u_2 \end{bmatrix} = \begin{bmatrix} \dfrac{Q_{f_1}}{1 + Q_{b_1}} \\ \dfrac{Q_{f_2}}{1 + Q_{b_2}} \end{bmatrix} (y_r - y + G_1 u_1 + G_2 u_2) + \begin{bmatrix} \dfrac{\tilde{P}_1}{1 + Q_{b_1}} \\ \dfrac{\tilde{P}_2}{1 + Q_{b_2}} \end{bmatrix} u_{1_r}$$

This means that

$$\begin{bmatrix} Q_1 \\ Q_2 \end{bmatrix} = \begin{bmatrix} \dfrac{Q_{f_1}}{1 + Q_{b_1}} \\ \dfrac{Q_{f_2}}{1 + Q_{b_2}} \end{bmatrix} \text{ and } \begin{bmatrix} P_1 \\ P_2 \end{bmatrix} = \begin{bmatrix} \dfrac{\tilde{P}_1}{1 + Q_{b_1}} \\ \dfrac{\tilde{P}_2}{1 + Q_{b_2}} \end{bmatrix}$$

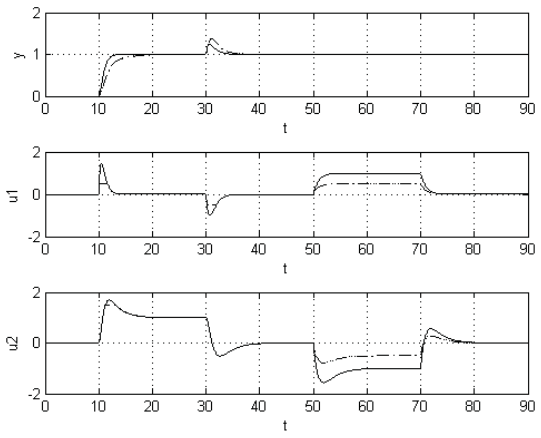


Fig. 10 Example 1: Response of IMC scheme with no saturation (solid), $\lambda = 0$ (dashed) and $\lambda = 1$ (dotted) for a unit changes in y_r at t=10, d at t=30, u_{1_r} at t=50 and t= 70.

From Zheng et al. (1994), Q_f can usually be chosen by the following relationship:

$$Q_f = \lambda Q(\infty) + (1 - \lambda)Q$$

- Condition 1: when $\lambda = 0$, then $Q_f = Q$ and from (7) $Q_b = 0$
- Condition 2: when $\lambda = 1$, then $Q_f = Q(\infty)$ and from (7) $Q_b = (Q(\infty) - Q)/Q$

The classical IMC structure as shown in Fig. 3 corresponds to the condition where $\lambda = 0$ and the modified IMC structure given by the block diagram in Fig. 9 corresponds to $\lambda = 1$. For condition 2 above, Q_{f_1} cannot be chosen to be $Q(\infty)$ because for the mid-ranging control problem it is required that $Q_{f_1}(0) = 0$. To ensure that $Q_{f_1}(0) = 0$, Q_f can be chosen such that,

$$Q_{f_1} = Q_1(\infty)\frac{s}{s + p_1}$$

where p_1 is a pole of Q_1.

Q_{f_2} can simply be chosen as $Q_{f_2} = Q_2(\infty)$. However, if Q_2 is strictly proper then Q_2 can be defined as:

$$Q_2 = Q_{2_b} \times Q_{2_s}$$

where Q_{2_b} is bi-proper and Q_{2_s} is strictly proper so that $Q_{f_2} = Q_{2_b}(\infty) \times Q_{2_s}$.

Fig. 10 shows the response of the system in example 1 defined in Section 6 when u_1 and u_2 saturate. Responses for both the classical IMC structure (Fig. 3) and the modified IMC structure (Fig. 9) are considered. For this example the response when $\lambda = 0$ and $\lambda = 1$ is similar because $Q_{f_1} = Q_1$.

When u_1 saturates, output tracking is still achieved by the slow actuator. The control action of u_2 is the same when u_1 saturates as in the unsaturated case. When u_2 saturates however, the fast actuator compromises between input and output tracking. Therefore the classical IMC structure offers acceptable performance during saturation. In some cases, the modified IMC structure offers even better performance with saturation of u_1 and u_2.

8. CONCLUSION

Many mid-ranging designs have been proposed and of them, MVPC and Direct Synthesis are most appropriate to address the ad hoc nature of conventional mid-ranging tuning procedures. MVPC works sufficiently especially when restricted to conventional mid-ranging structure and PID control. However, when not restricted, the Direct Synthesis (Henson et al., 1995) and IMC approaches give better performance. The IMC mid-ranging design presented in this paper gives the same improved performance over MVPC as Direct Synthesis. Moreover, this approach gives insight to the design trade-offs of MVPC and Direct Synthesis by emphasis on bandwidth considerations. An added benefit of the IMC approach is the ability to integrate anti-windup to mid-ranging control. Other mid-ranging approaches require additional control blocks and further modifications to achieve acceptable performance under saturation of the inputs. IMC mid-ranging is already a natural structure for anti-windup.

REFERENCES

ALLISON, B. J. & ISAKSSON, A. J. (1998) Design and performance of mid-ranging controllers. *Journal of Process Control,* 8**,** 469-474.

ALLISON, B. J. & OGAWA, S. (2003) Design and tuning of valve position controllers with industrial applications. *Transactions of the Institute of Measurement and Control,* 25**,** 3-16.

HAUGWITZ, S., KARLSSON, M., VELUT, S. & HAGANDER, P. (2005) Anti-windup in mid-ranging control. *Fourty-fourth IEEE Conference on Decision and Control, and the European Control Conference 2005.* Seville, Spain.

HENSON, M. A., OGUNNAIKE, B. A. & SCHWABER, J. S. (1995) Habituating Control Strategies for Process-Control. *AIChE Journal,* 41**,** 604-618.

MORARI, M., ZAFIRIOU, E. (1989) *Robust Process Control*, Prentice-Hall, Inc.

PRETT, D. M. & GARCIA, C., E. (1988) *Fundamental Process Control,* Stoneham, Butterworth Publishers.

ZHENG, A., KOTHARE, M. V. & MORARI, M. (1994) Antiwindup Design for Internal Model Control. *International Journal of Control,* 60**,** 1015-1024.

Robust optimization-based multi-loop PID controller tuning: A new tool and an industrial example

Michael Harmse*, Richard Hughes, Rainer Dittmar*****
Harpreet Singh* and Shabroz Gill*

**IPCOSAptitude Ltd., Cambridge, United Kingdom*
(e-mail: info@ipcosaptitude.com)
***SABIC UK Petrochemichals Ltd., Wilton, Middlesbrough, United Kingdom*
**** West Coast University of Applied Sciences, Heide, Germany*
(e-mail: dittmar@fh-westkueste.de)

Abstract: Modern process plants are highly integrated and as a result, decentralized PID control loops are often strongly interactive. The currently used sequential tuning approach is not only time consuming, but does also not achieve optimal performance of the inherently multivariable control system. This paper describes a method and a software tool which allows a control engineer to calculate optimal PID controller settings for multiloop systems. It is based on the identification of a state space model of the multivariable system, and it uses constrained nonlinear optimization techniques to find the controller parameters. The solution is tailored to the specific control system and PID algorithm to be used. The methodology has been successfully applied in several industrial advanced control projects. The tuning results which have been achieved for interacting PID control loops in the stabilizing section of an industrial Gasoline Treatment Unit at SABIC Petrochemicals are presented.

Keywords: PID controller tuning, multi-loop control, decentralized control system design, nonlinear optimization, genetic algorithm, multivariable system identification.

1. INTRODUCTION

One of the most important challenges facing the process industry today is optimizing the operation of complex units, without compromising the safety and integrity of the process equipment. Process complexity has increased significantly over the past two decades due to increased level of heat integration and use of recycle streams. In addition, the need for increased process flexibility to deal with changing raw materials and alternate energy sources, as well as the need to adapt quickly to fluctuating throughput and quality targets, often means that the process dynamics will vary significantly over time and with operating point. The basic control layer of process plants almost always consists of a large number of decentralized SISO PID controllers, although this approach is intrinsically inadequate for multivariable processes. Due to the situation described above, the interactions between these controllers are becoming more important, and tuning these control loops for good performance and adequate robustness is a challenging task.

The industrial practice of PID controller tuning is still dominated by manual trial-and-error tuning. If tuning rules are used at all, it's the "classical" ones like Ziegler-Nichols or Chien-Hrones-Reswick which are based on simplified first order plus dead time (FOPDT) process models and do not consider stability robustness issues, therefore often being not adequate in modern process units with more complex dynamics and nonlinearity. In addition, many tuning rules assume that all PID controller equations work as described in the textbooks, when in fact there is substantial variation between the different vendors. In contrast, different PID controller structures result due to use of either the parallel or the serial form, using the control error or the PV by the Proportional (P) and Derivative (D) terms, and many other quirks like alternative implementations of the derivative filters. Tuning SISO PID controllers in a multivariable environment is usually done in a time-consuming sequential and iterative way, starting with the most important loops, and heuristic detuning in case the interactions are significant.

For a long time, vendors of automation systems such as Distributed Control Systems (DCS) and Programmable Logic Controllers (PLC) have been offering PID self-tuning functionality (tuning on demand). Unfortunately, they have only found limited application. This is also true for model based PID controller tuning software provided by the same or third-party vendors. Moreover, in most cases these tools are restricted to single loop tuning applications, and do not support multi-loop tuning (Li et al., 2006, Espinosa Oviedo et al., 2006 and Zhu, 2004).

The design of interacting PID controllers in a multivariable environment is not a new topic in the process control literature. At least three research directions can be identified: (1) reduction of controller interactions by proper MV-CV pairing, (2) design of decoupling networks and (3) consideration of MIMO interactions in decentralized controller tuning. In this paper, only the third direction is relevant. Several methods have been developed, Luyben's

BLT method being the most popular one (Monica et al., 1988). Here, the individual PI loops are first tuned by the Ziegler-Nichols rules independently. Then, a detuning factor is calculated which assures a certain stability margin for the controlled MIMO system. All individual controller gains are divided by this factor, and the reset times are multiplied by it. The price to be paid for the reduced interaction is a more sluggish behaviour of PI loops. Other methods include the sequential loop closing approach (Hovd and Skogestad, 1994), the independent design method (Hovd and Skogestad, 1993) and the multivariable generalization of the relay-feedback self-tuning method (Halevi et al. 1997). For a discussion of these methods the reader is referred to (Chen and Seborg, 2003).

This paper introduces a new method and a software tool "AptiTune™" for the calculation of optimum PID controller settings in a multivariable system (multivariable loop tuning). The method consists of several steps. First, a set of Finite Impulse Response (FIR) models of the open-loop MIMO plant is being identified and approximated by a reduced-order state space model. In a second step, optimal parameters for the decentralized PID controllers are calculated using constrained optimization. Finally, the setpoint tracking, disturbance rejection and noise attenuation behaviour of the controlled system is simulated.

It was the aim of the development to come up with a software tool which is based on recent identification and control developments, but which does not require in-depth knowledge of identification and control theory by the average user. Furthermore, the optimization solution is tailored to the specific target automation system, e.g. the particular DCS or PLC which is used for control purposes.

The remainder of the paper is organized as follows: In section 2, the identification and optimal tuning methods will be described together with the "AptiTune™" software tool. Section 3 presents some results of multiloop tuning in the stabilizer section of an industrial Gasoline Treatment Unit (GTU). The retuning of the PID controllers was one of the first steps of an advanced control project, which also included the design and commisioning of an MPC controller.

2. METHOD AND TOOL FOR MULTILOOP TUNING

2.1 Identification of the MIMO process model

The first step of model based multiloop tuning is to develop a dynamic model of the multivariable process with n inputs and n outputs, the outputs (u_i) and process variables (y_i) of the PID controllers shown in Fig. 1.

Our preferred approach is to switch all PID controllers to be tuned into manual mode whenever possible and to perform a series of output steps of different duration and amplitude. According to our experience, four to six steps with duration varying between 10% and 100% of the desired closed-loop settling time are usually sufficient. If a test signal generator is available, PRBS (pseudo-random binary sequence) or GBN

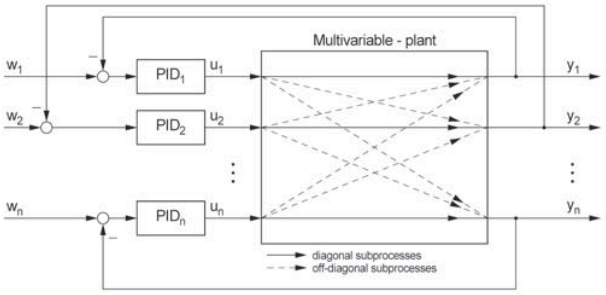


Fig. 1: Decentralized multiloop PID control system

(generalized binary noise), then an automated test may be used as an alternative. Both types of plant tests can be performed in sequential or in time-saving simultanous mode.

If one or more PID controllers cannot be switched to manual mode, then the loop can be kept in automatic mode and multiple setpoint steps can be made. The Projection Method described in (Forsell and Ljung, 2000) can then be used.

After pre-processing the raw test data (detection/rejection of outliers, filtering, decimation, cutting out periods of bad data etc.), the parameters of a MIMO FIR model

$$\vec{g}_{ij} = \left[g_{ij}(0), g_{ij}(1), g_{ij}(2), \ldots, g_{ij}(n_M) \right] \qquad i, j = 1 \ldots n \quad (1)$$

are estimated by least squares regression. The user should specify a-priori knowledge such as zero gain, known dead time or integrating behaviour of subprocesses. Although FIR models are estimated, the results are presented as Finite Step Response (FSR) models for easier visualization and understanding. The "AptiTune™" software tool also supports the import of FSR models created by identification tools from MPC packages, but also allows the user to specify a transfer function matrix.

In the next step, the MIMO FIR model is approximated by a linear state-space model of the form

$$\begin{aligned} \dot{\boldsymbol{x}}(t) &= \boldsymbol{A}\,\boldsymbol{x}(t) + \boldsymbol{B}\,\boldsymbol{u}(t) \\ \boldsymbol{y}(t) &= \boldsymbol{C}\,\boldsymbol{x}(t) \end{aligned} \qquad (2)$$

This approximation is not based on the raw or preprocessed plant test data, but on a model-to-model fit. To remove noise and cycles from the FIR model, it can first be smoothed using a central average filter. The state-space model is constructed using the singular value decomposition (SVD) model reduction technique (Maciejowski, 1989). While creating the state-space model, the diagonal model curves are given more preference than the off-diagonal models. As a result, diagonal models normally have higher order than the off-diagonal ones and consequently fit the original FIR model curves more accurately. The step responses calculated based on the stat-space models are graphically displayed.

If it is possible to do a closed-loop step test (or if historical data contain a clear SP step), a practical way of validating the process model is to simulate the closed loop behaviour of the control system with the actual PID controller parameters

currently entered on the DCS , and to compare the simulation results with plant data. If the observed responses are similar to the simulated responses, then we can conclude that the model is sufficiently accurate for loop tuning purposes.

2.2 Calculation of optimal PID controller parameters

The PID controller parameters (controller gains $K_{c,i}$, reset times $T_{r,i}$ and derivative time constants $T_{d,i}$) are calculated solving numerically the nonlinear constrained optimization problem

$$\min_{K_{Pi},T_{Ni},T_{Vi}} \quad J$$
$$g_j(K_{Pi},T_{Ni},T_{Vi}) \le 0 \quad i=1...n,\ j=1...m \tag{3}$$

where J denotes the objective function and g_j are constraints. The objective function J is a weighted sum of three terms $J = J_1 + \alpha J_2 + \beta J_3$ which assess different aspects of the control loop performance. The first part $J_1 = \int_0^{T_f} \left| y(t) - y_r(t) \right| dt$ refers to the Integrated Absolute Error (IAE) criterion for setpoint tracking. Here, the error is defined as the difference between the PV and a user-defined first order reference trajectory $y_r(t)$ connecting the actual PV and the setpoint. By specifying the time constant of the trajectory, the user can affect the speed of the response to setpoint changes. The second part $J_2 = \int_0^{T_f} \left| w(t) - y(t) \right| dt$ denotes the IAE for an input step disturbance. Finally, the third term $J_3 = \int_0^{T_f} \left| \Delta u(t) \right| (t)$ reflects the control effort. By setting the weighting coefficients α and β, the user can balance a compromise between the different performance objectives. Another design parameter allows the user to weight the performance of the n SISO control loops against the necessary degree of decoupling between them.

For each control loop, the user can specify one or more inequality constraints $g_j \le 0$ from the following list: maximum OP deviation after setpoint changes, maximum overshoot, minimum damping ratio, maximal noise amplification, process gain and deadtime margins, maximum/minimum limits of the controller parameters. For level buffering controllers, the maximum setpoint deviation and the minimum return time after a level disturbance can be specified. By careful specification of the constraints, the user can tailor the tuning to process-specific requirements.

For starting the numerical optimization, initial controller parameter values have to be selected. For this purpose, the user can choose to use the actual DCS values or values calculated by the Cohen-Coon tuning rule (for individual controller tuning assuming a SISO model). The degree of difficulty of the nonlinear constrained optimization problem depends on the number of controllers involved, the order of the process model, and the number and nature of the inequality constraints. In general, non-convexity and local minima can occur. Therefore, several search algorithms have been implemented, including a brute force global search in the entire parameter space, a genetic algorithm (both intended for initialization), and a generalized gradient algorithm (Vlachos et al., 2000).

In contrast to some PID controller tuning software available mainly for teaching and training purposes, the "AptiTune™" tool not only calculates "generic" PID controller parameters, but parameters for a specific realization of the PID controller equation for specific commercial control system hardware. The user can select between different control algorithms of widespread DCS systems such as Honeywell, Emerson DeltaV, Foxboro I/A, ABB and several others. For example, six different versions of the PID algorithms are available for the Emerson DCS, for which the optimization results may be quite different. Optimal controller parameters can of course also be calculated for P, I only and PI controllers.

After the optimizer has converged and optimal controller parameters have been found, the design process will be finished by simulation of the dynamic behaviour of the control system. It is useful to study different scenarios: setpoint tracking, input disturbance rejection, and noise attenuation. The "integrity" of the controlled MIMO system should be studied as well, i.e. the behaviour of the controlled system if one or more controllers are in manual mode, or if components like final actuators fail. Finally, the robustness against plant-model mismatch should be evaluated. For this purpose, robustness plots such as in Fig. 2 are helpful. It shows the purple stability limit in a process gain ratio/dead time graph, and the stability region for the minimum required combined gain and deadtime margins as the red polygon (left hand plot, lower left hand corner).

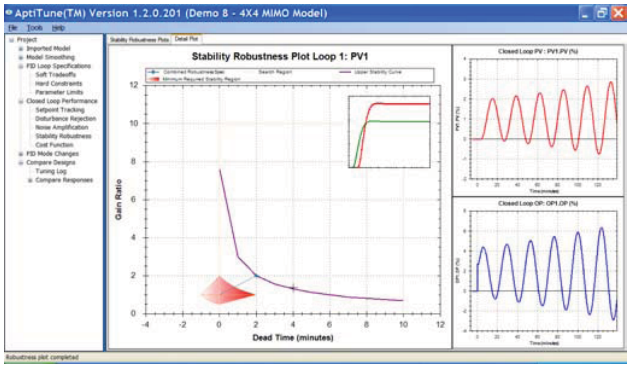


Fig. 2: Robustness plot

3. INDUSTRIAL EXAMPLE

The method and software tool described above have been used successfully in a number of advanced control projects. A good example is the stabilizing section of a GTU process, where improving the PID controller tuning was a prerequisite for successful MPC design and implementation.

The Process and Instrumentation Diagram for the GTU process is shown in Fig. 3. Although the overall system is (8x8), it was possible to decompose it into a (2x2) system on

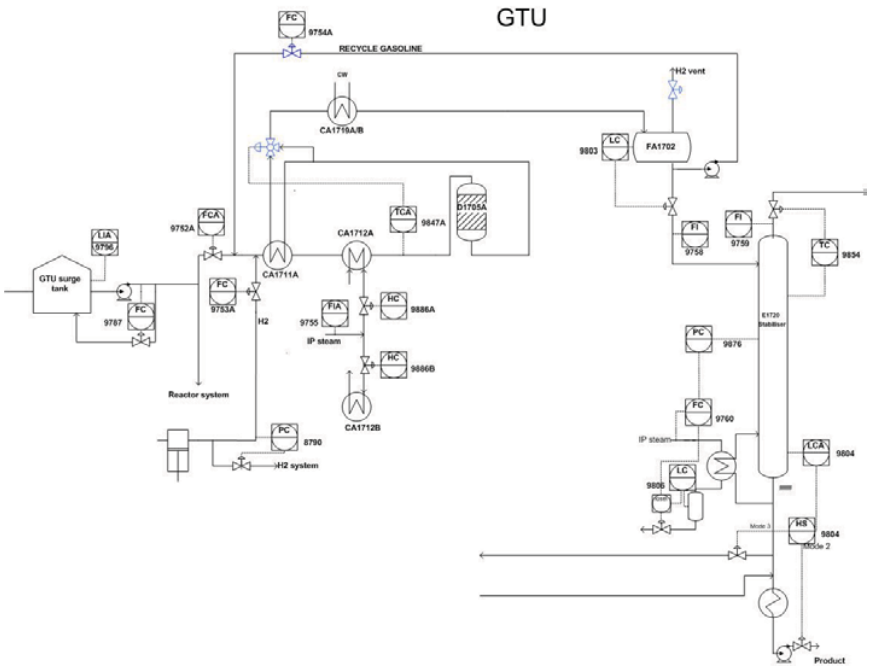


Fig. 3: P&I diagram of the GTU process

the E20 column and (6x6) for the rest of the unit. Due to limited space, the results for the (2x2) system are presented.

The objective of the E20 stabilizer column (on the right hand side of the P&ID) is to remove hydrogen and methane dissolved in the petrol (mostly C5) stream. The column is essentially a degassing drum with trays for improved separation. The current PID control scheme is somewhat unconventional in that the PID loop pairing is the "wrong" way round:

- A tray temperature close to the top of the column is controlled (even though the product specification is on the bottoms stream) by using the overhead vent valve for temperature control. There is no reflux drum, and feed comes in close to the top, providing the internal reflux stream.
- The column pressure controller cascades to the steam flow SP on the reboiler.

One reason for the unconventional PID loop pairing is that controlling pressure with the reboiler duty ensures that the pressure is less likely to go high and lift the safety valve. If the PID was paired the other way around, there is a chance for the pressure to go high if the overhead valve saturates before the column runs out of reboiler duty. Since the degree of interaction between the two loops is quite strong, the pressure loop will go unstable if either the feed drum LIC or the TIC approached unstable (even though there is nothing wrong with the tuning of the PIC). The drum level controller sets the valve position directly without the benefit of a cascaded flow controller. Any change in either upstream or downstream pressure affect the feed flow PV, which then affects the temperature and pressure in the E20 column. Due to the heavy coupling between the three loops, it is not really feasible to tune one loop without due consideration of the remaining two control loops.

Best results were obtained by step testing condensate flow setpoint and vent valve, fitting a 2x2 model and then calculating moderately fast but well damped tuning that takes the strong off-diagonal interaction into account. We tuned it for fast SP tracking but high robustness margins. The standard deviations of the two PVs are now substantially better than before and that both loops are tracking SP well.

The gain for the tray temperature vs. vent valve varies very substantially depending on operating point (from almost zero when the top of the column is perfectly pure, to almost infinite when the column goes rapidly off-spec). This value represents the best model we could identify by keeping the tray temperature in the correct range with the column pressure at the nominal operating point. The dynamic model is shown in Fig. 4.

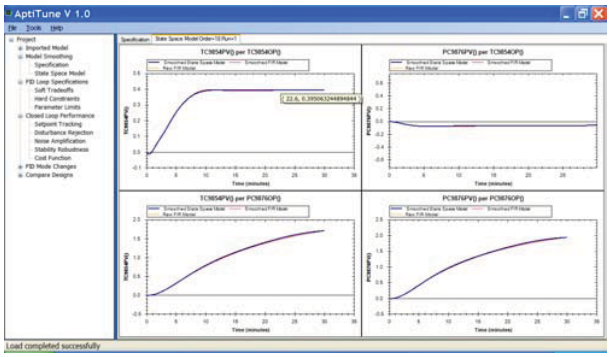


Fig. 4: Step responses of the dynamic model
(Legend: TC.PV=Tray 6 Temperature; PC=Column Pressure; TC.OP=Vent Valve; PC.OP=Steam Flow SP)

Note that the total state-space model order is 18. Fig. 5 shows the optimized PV responses for a step change in both SPs:

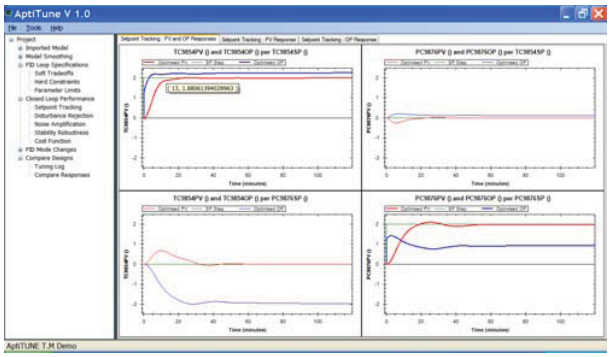


Fig. 5: Response of the 2x2 system to SP step changes
(Legend: PV is shown in red, OP in blue)

Note that PV overshoot is very low and that damping is exceptionally good. The OP value for the vent has a peak value that is almost the same as the steady state value, and the loop has about the same rise time in closed loop as compared to open loop (a speed-up factor of about 1x). The pressure loop is about 2x faster in closed loop compared to open loop. The load disturbance response is shown in Fig. 6:

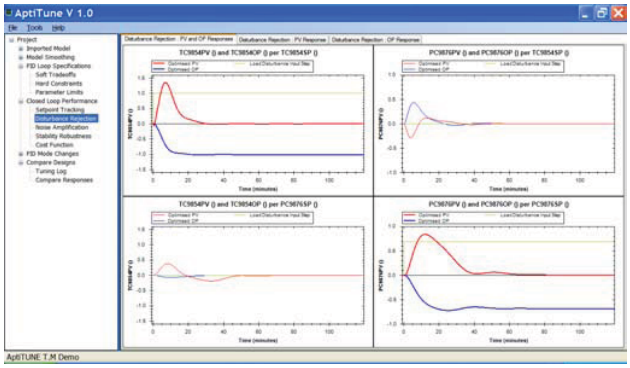


Fig. 6: Response of the 2x2 system to load disturbances

Note that the PV and OP responses have very good damping with peak OP values that are very similar to the steady state values. This will ensure exceptionally good damping on the actual process unit even when the process gain varies significantly. Gain and dead time (stability) margins are very good, see Fig. 7:

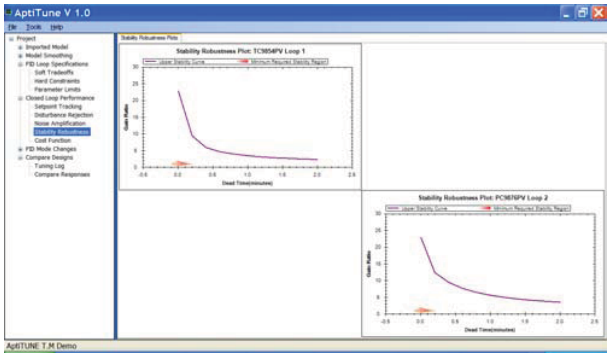


Fig. 7: Robustness plot (gain and deadtime stability margins)

A sensitivity analysis on the tray temperature loop shows that the gain of the process will have to increase by 3x AND the dead time will have to increase by another 12 seconds before the damping of the loops is unacceptable. Instability sets in at a process gain increase of more than 20x. A dead time error of more than 2 minutes is needed to reach instability, and there is no process mechanism for this to occur while the TIC is in the active range. These margins are exceptionally safe.

In order to compare the performance of the loops before and after retuning, we collected a week of normal operating data before we arrived on site, and one week of normal operating data after the re-tuning work was concluded. From these large data sets, we then calculated histograms to show the distribution of the control error (SP-PV). For process reasons, we wanted the loops to be robust and to be able to withstand changes in process dynamics. As a result, some loops were intentionally slowed down, and of course, their probability distributions will be wider than before. However, this compromise is all for a good cause as it will ensure that the loops remain operational for the years to come.

The performance of the pressure control loop PCA9876 is compared in Fig. 8:

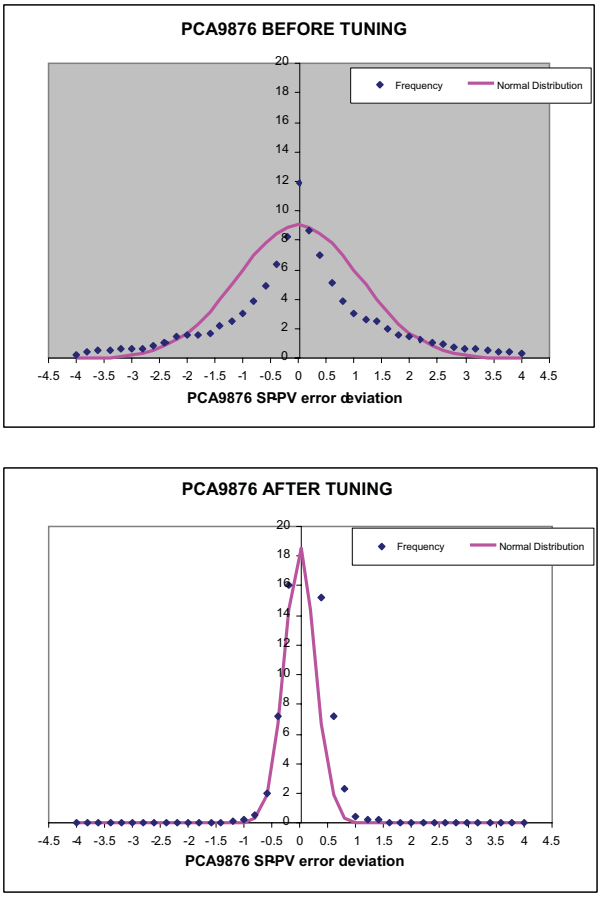


Fig. 8: Control error histograms for PCA9876 before and after retuning

It is clear from the two histograms shown above that the variability in the PV has reduced by about a factor of 4x. This is a big improvement in performance, yet this could be accomplished without compromising the robustness characteristics of the loop. The pink trace shows the best fit for a normal (Gaussian) distribution to the data, assuming a zero mean value. The estimated standard deviation σ reduced from 1.1 to 0.28.

The histograms for temperature loop TC9854 are compared in Fig. 9:

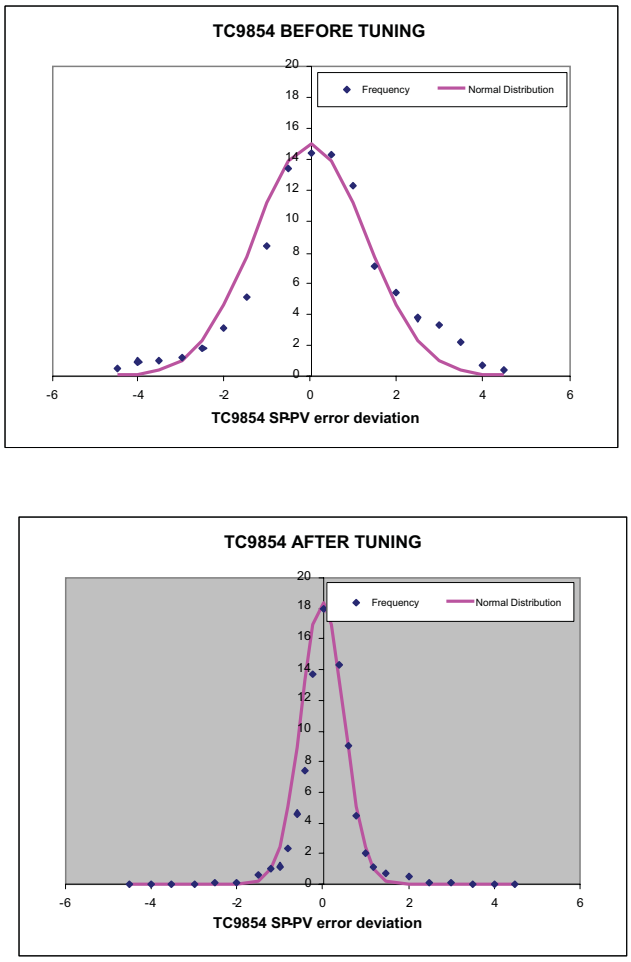


Fig. 9: Control error histograms for TC9854 before and after retuning

The standard deviation reduced from 1.3 down to 0.5, so by a factor of almost 3x. To be honest, these good results are partly due to the very slow initial tuning of some loops we found at the beginning of the project.

4. CONCLUSIONS

The following conclusions can be made. A MIMO model-based approach can be used to successfully tune multiple PID loops that interact strongly. If the open loop model is moderately accurate, then "one-shot" tuning is achievable and the simulated and observed OP and PV responses will be almost identical. The use of sufficiently large gain and dead time robustness margins ensures that the loop will remain stable and well damped even if the process is strongly nonlinear. This also helps to protect us against inaccurate model identification results. The ability to impose hard constraints on damping ratio, maximum PV overshoot, and the maximum OP value means we can ensure that the final design is safe from a process point of view. PID tuning rules cannot achieve this.

REFERENCES

Chen, D., Seborg, D.E. (2003). Design of decentralized PI control systems based on Nyquist stability analysis. *Journal of Process Control* 13(1), 27 – 39.

Espinosa Oviedo, J.J., Boelen, T., van Overschee, P. (2006). Robust advanced PID control. *IEEE Control Systems Magazine* 26(1), 15 – 19.

Forssell, U., Ljung, L. (2000). A projection method for closed-loop identification. *IEEE Trans. On Automatic Control* 45(11) 2101 – 2106.

Halevi, Y., Palmor, Z.J., Efrati, T. (1997). Automated tuning of decentralized PID controllers for MIMO processes. *Journal of Process Control* 7(2), 119 – 128.

Hovd, M., Skogestad, S. (1993). Improved independent design of robust decentralized controllers. *Journal of Process Control* 3(1), 43 – 51.

Hovd, M., Skogestad, S. (1994). Sequential design of decentralized controllers. *Automatica* 30(10), 1601 – 1607.

Li, Y., Ang, K.H., Chong, G. (2006). Patents, software and hardware for PID control. *IEEE Control Systems Magazine* 26(1), 42 – 54.

Maciejowski, J.M. (1989). *Multivariable Feedback Design*. Addison-Wesley

Monica, T.J., Yu, C.-C., Luyben, W.L. (1988). Improved multiloop single-input, single-output (SISO) controllers for multivariable processes. *Ind. Eng. Chem. Res.* 27, 969 – 973.

Vlachos, C., Williams, D., Gomm, J.B. (2000). Genetic approach to decentralized PI controller tuning for multivariable processes. *IEE Proceedings. Control Theory and Applications* 146(1), 58 – 64.

Zhu, Y. (2004). Robust PID tuning using closed-loop identification. Preprints of the International Symposium on Advanced Control of Chemical Processes ADCHEM 2004. vol. 1, 165 – 170

Auto-tuned Predictive Control Based on Minimal Plant Information

G. Valencia-Palomo * **J.A. Rossiter** *

* *Department of Automatic Control and Systems Engineering, University of Sheffield, South Yorkshire, U.K. S1 3JD. (e-mail: g.valencia-palomo@shef.ac.uk, j.a.rossiter@shef.ac.uk).*

Abstract: This paper makes two key contributions. First there is a definition and implementation of a novel auto-tuned predictive controller. The key novelty is that the modelling is based on relatively crude but pragmatic plant information. Secondly, the paper tackles the issue of availability of predictive control for low level control loops. Hence the paper describes how the controller is embedded in an industrial Programmable Logic Controller (PLC) using the IEC 1131.1 programming standard. Laboratory experiment tests were carried out in two bench-scale laboratory systems to prove the effectiveness of the combined algorithm and hardware solution. For completeness, the results are compared with a commercial PID controller (also embedded in the PLC) using the most up to date auto-tuning rules.

Keywords: Predictive control, auto-tuning, programmable logic controller, IEC-1131.1.

1. INTRODUCTION

Control design methods based on the predictive control concept have found wide acceptance in industry and in academia, mainly because of the open formulation that allows the incorporation of different types of models of prediction and the capability of constraint handling in the signals of the system.

Model predictive control (MPC) has had a peculiar evolution. It was initially developed in industry where the need to operate systems at the limit to improve production requires controllers with capabilities beyond PID. Early predictive controllers were based in heuristic algorithms using simple models. Small improvements in performance led to large gains in profit. The research community has striven to give a theoretical support to the practical results achieved and thus the economic argument, predictive control has merited large expenditure on complex algorithms and the associated architecture and set up times. However, with the perhaps notable exception of Predictive Functional Control (PFC) (Richalet, 1993), there has been relatively little penetration into markets where PID strategies dominate, and this despite the fact that predictive control still has a lot to offer in the SISO domain because of its enhanced constraint handling abilities and the controller format being more flexible than PID. The major obstacles cost, complexity and the algorithm not being available in the off the shelf hardware most likely used for local loop control.

Some authors have improved the user-friendliness (complexity) of MPC software packages available for high level control purposes (Froisy, 2006; Zhu *et al.*, 2008). Nevertheless, they have the same implementation drawback in that the development platform is a stand-alone computer running under Windows® OS. Furthermore, these packages involve complex identification procedures which thus requires the control commissioning to be in the hands of a few skilled control engineers; ownership by non control experts is an impediment for more widespread utilization.

Some early industrial work (Richalet, 2007) has demonstrated that with the right promotion and support, technical staff are confident users of PFC where these are an alternative to PID on a standard PLC unit. Technical staff relate easily to the tuning parameters which are primarily the desired time constant and secondly a coincidence point which can be selected by a simple global search over horizons choices. Because PFC is based on a model, the controller structure can take systematic account of dead-times and other characteristics, which are not so straightforward with PID. Also constraint handling can be included to some extent by using predicted violations to trigger a temporary switch to a less aggressive strategy.

The vendors conjecture is that PFC was successfully adopted because of two key factors: first there is effective support in technician training programmes (get it on the syllabus) and second the algorithm is embedded in standard PLC hardware they encounter on the job, thus making it easily accessible (and cheap). However, despite its obvious success academia has shied away from the PFC algorithm because its mathematical foundations are not as systematic or rigorous as other approaches; the performance/stability analysis is primarily an a *posteriori* approach as opposed to the a *priori* one more popular in modern literature. So there is a challenge for the academic community to propose more rigorous but nevertheless intuitive and simple algorithms which could equally be embedded in cheap control units.

On the other hand, in recent specialized conferences authors are often focussing on the level of rigor required in the modelling and tuning procedure for different cases (Morari *et al.*, 2008). However, accessibility and useability

in such a mass market may require different assumptions from those typically adopted in the literature; specifically much less rigor and more automation in the modelling will be essential.

Hence, the first objective of this paper is to develop an auto-tuned MPC controller based on minimal plant information which would be available from staff at technician level only who may be responsible for maintaining and tuning local loops. Secondly, the paper aims to demonstrate how an MPC algorithm, using this model information, can be embedded in a commercial PLC (Valencia-Palomo and Rossiter, 2008); this paper gives some extensions to that developments in (Valencia-Palomo *et al.*, 2008) and of particular interest to readers will be the incorporation of systematic constraint handling within the PLC unit. A final objective is to contrast the auto-tuned MPC with a commercial PID controller in order to show that the MPC is a practical (available and same cost) alternative to PID for local loops.

The paper is organized as follows: Section 2 outlines the controllers and the auto-tuning rules, Section 3 describes the implementation of the controllers in the target hardware, Section 4 presents the simulation results on real hardware and finally in Section 5 are the conclusions and future work.

2. THE CONTROLLERS

This section outlines the auto-tuning rules and modelling assumptions for the MPC and PID strategies adopted. We note that the auto-tuning rules are only applicable to stable systems so discussion of unstable systems is deferred for future work.

2.1 Modelling assumptions

If anything, this paper is more generous with the auto-tuned PID than the MPC because it allows the PID algorithm a large quantity of measurement data and the ability to dither the input substantially during tuning to extract the required information. Moreover, the complexity of this algorithm means that the modelling is done offline. This decision was taken to give a stiff test for the auto-modelled/tuned MPC algorithms.

For MPC we provide crude modelling information only, for instance as could be provided by a technician or plant operator but specifically avoiding the use of a rigorous least squares model estimator which could be expensive if required for large numbers of loops and impractical to put on the PLC unit. The technician should provide estimates of behaviour as compared to standard second order characteristics: rise-time, settling time, overshoot, steady-state gain and dead-time. From this data an approximate second order model with dead-time is determined [1].

2.2 Design point, auto-tuning and constraint handling for PID

A novel auto-tuned PID controller as described in (Clarke, 2006; Gyöngy and Clarke, 2006) is used. A schematic

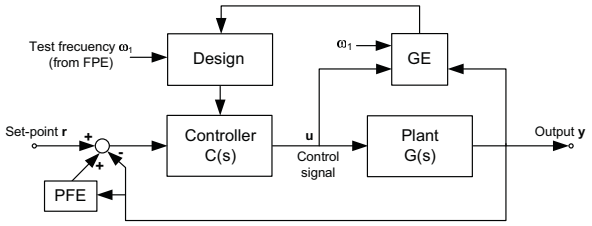


Fig. 1. Schematic diagram of the auto-tuning PID.

diagram of the system is shown in Fig. 1. The objective is to adapt the controller so as to achieve a carefully chosen design point on the Nyquist diagram.

The key components are phase/frequency and plant gain estimators (PFE, GE), described in detail in (Clarke, 2002). In essence a PFE injects a test sinewave into a system and continuously adapts its frequency ω_1 until its phase shift attains a desired value θ_d (in this case the design point). Also forming important part of the tuner, but not shown in Fig. 1, are variable band-pass filters (VBPF) at the inputs of the PFE and GE. These are second-order filters centered on the current value of the test frequency. They are used to isolate the probing signal from the other signals circulating on the loop (such as noise, set-point changes and load disturbances).

The algorithm is initialized using a first-order/dead-time (FODT) approximation $G_a(s)$ for the plant, obtained from a simple step test. The initialization involves the computation of suitable values for the parameters associated with the GE, PFE and the controller.

The controller is based on a design point in the Nyquist diagram. This design point is chosen to obtain the desired closed loop behavior, i.e. rise time, damping value, settling time. In this case, the desired damping value of 0.5 for all the systems is chosen. From this desired damping value, the variables for all the auto-tuning process are obtained as is shown in (Clarke, 2006; Gyöngy and Clarke, 2006).

The PID design does not take explicit account of constraints and thus ad hoc mechanisms are required. Typically input saturation with some form of anti-windup will be used but state constraints are not considered; this is a weakness.

2.3 Basic assumptions for MPC

For the purpose of this paper almost any conventional MPC algorithm can be deployed as the main distinguishing characteristic, **with sensible tuning**, is the model. Hence, assume that the MPC law can be reduced to minimising a GPC [2] cost function of the form:

$$J = \sum_{j=1}^{H_P} \|\hat{\mathbf{y}}(k+j|k) - \mathbf{w}(k+j|k)\|^2 + \sum_{j=1}^{H_C} \|\Delta\mathbf{u}(k+j|k)\|_\lambda^2 \tag{1}$$

where the second term in the eq. (1) is the control effort and λ is the weighting sequence factor. The reference trajectory $\mathbf{w}(k)$, is the desired output in closed loop of the system and is given by:

[1] We accept that for more complex dynamics a slightly more involved procedure may be required.

[2] To simplify some algebra compared to dual-mode approaches, e.g. (Rossiter *et al.*, 1998).

$$\mathbf{w}(k+i|k) = \mathbf{s}(k+i) - \alpha^i [\mathbf{s}(k) - \mathbf{y}(k)]; \ 1 \leq i \leq H_P \quad (2)$$

where $\mathbf{s}(k)$ is the set-point and α determines the smoothness of the approach from the output to $\mathbf{s}(k)$. The objective (1) can be expressed in more compact form in terms of the predicted output:

$$\min_{\Delta \mathbf{u}} \mathbf{J}(\Delta \mathbf{u}) = \frac{1}{2} \Delta \mathbf{u}^T \mathbf{H} \Delta \mathbf{u} + \mathbf{f}^T \Delta \mathbf{u} + \mathbf{b} \quad (3)$$

$$s.t. \qquad \mathbf{R} \Delta \mathbf{u} \leq \mathbf{c} \quad (4)$$

where $\Delta \mathbf{u}$ is the vector of future inputs increments and the other matrix details are omitted for brevity but available in standard references, e.g. (Maciejowski, 2002; Rossiter, 2003; Camacho and Bordons, 2004).

The tuning parameters are usually taken to be the horizons H_P, H_C and weights λ. However, more recent thinking suggests that H_P should be larger than the settling time, H_C is typically 2 or 3 (for practical reasons rather than optimality which requires higher values) and λ becomes the major tuning parameter, albeit some may argue a poor mechanism for tuning. The parameter α will also have a substantial impact but is rarely discussed except in PFC approaches.

2.4 Constraint handling for MPC

The systems considered in this paper are stable, therefore in the absence of output constraints, for a reachable set point the system will only violate the constraints in presence of disturbances or overshoots derived from set point changes. In practice, one may not be able to program a complete QP solver, so a sensible way of handling constraints is to interpolate two control laws (Rossiter and Grieder, 2005), one with good performance (e.g. $\Delta \mathbf{u}_{fast}$) and one with good feasibility (e.g. $\Delta \mathbf{u}_{slow}$) , using:

$$\Delta \mathbf{u} = (1-\beta) \Delta \mathbf{u}_{fast} + \beta \Delta \mathbf{u}_{slow}; \quad 0 \leq \beta \leq 1 \quad (5)$$

The variable β is used to form the mix of fast and slow according to the predicted situation (if feasible $\beta = 0$). Hence, the optimization procedure reduces to simple linear program in one variable that is a set of inequality checks of the form:

$$\min \beta \ \ s.t. \ \ R_i \beta - c_i \leq 0, \quad i = 1, ..., H_C \quad (6)$$

Remark 1. If $\beta = 1$ and the constraints are still being violated, the inputs are saturated. Essentially this means the state is outside the maximal admissible set for the unconstrained control law designed for good feasibility. Such scenarios need more complex strategies not covered in this paper.

2.5 Simple auto-tuning rules for MPC

There are many alternatives for auto-tuning, some with better properties then used here are possible, but the authors felt this paper should initiate discussion with an industrial standard. Thus, the predictive control design and tuning procedure is described next.

For the MPC the prediction horizon H_P is chosen equal to the settling time plus H_C, with $H_C \ll H_P$. Assuming normalisation of input/output signals, $0.1 \leq \lambda \leq 10$, a form of global search can be used to settle on the 'best' parameters against some criteria, however, if we take the criteria to be the cost J of eq. (1) with $\lambda = 1$, then this fixes λ and H_C is chosen to be as large as possible [3]. The design response speed α_{fast} (for $\Delta \mathbf{u}_{fast}$) will be taken as half the open-loop time constant α_0 so that the controller has to deliver some extra speed of response as well as stability and offset free tracking; and α_{slow} (for $\Delta \mathbf{u}_{slow}$) will be taken as 0.95 to have a smooth close loop response and avoid overshooting in set point changes. Thus, the auto-tuning is fixed precisely by the model parameters and the technician role is only to provide best estimates of these parameters (in practice some iteration should take place).

Fig. 2. Allen Bradley PLC – SCL500 processor family.

3. IMPLEMENTATION OF THE ALGORITHMS ON A PROGRAMABLE LOGIC CONTROLLER

This section briefly introduces the PLC and the corresponding implementation of the controllers described in the previous section.

3.1 Allen Bradley — Rockwell Automation Inc. PLC

PLCs are by far the most accepted computers in industry which offer a reliable, safe and robust system; we will not revisit the reasons here. Nevertheless, normally their use is only to implement control sequences in open loop and/or different structures of PID controllers. For the purposes of this paper, the implementation is based on the family of SLC500 processors belonging to the Allen Bradley PLC systems, e.g. see Fig. 2.

The Allen Bradley set of PLC includes the facilities to be programmed in 3 of 5 languages in agreement with the IEC 1131.3 standard using Control Logix 5000^{TM} software programming package. Each of these allows for any combination of programming languages to be used for a single project. These three languages are: (i) *Ladder Diagram*, (ii) *Function Block Diagram* and (iii) *Structured Text*.

3.2 PID

The Control Logix 5000^{TM} software programming package also includes a function block to implement a PID controller. The PID function block is a professional development from Rockwell Inc. used in industry (with a PLC) to control a variety range of processes.

[3] For sensible sample times, a choice of $H_C \geq 5$ implies that this is approximately equivalent in behaviour to a dual-mode approach.

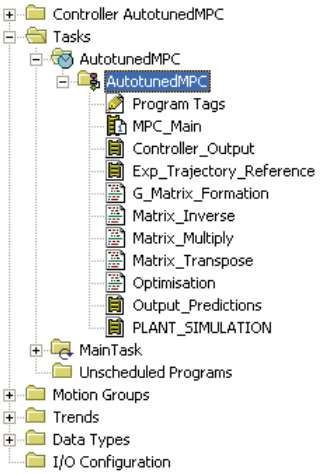


Fig. 3. Structure of the MPC algorithm in the target PLC.

The tuning of the PID is done off-line by the algorithm described in subsection 2.2. The obtained parameters are passed to the PID block before downloading the program to the PLC. As noted earlier, the PID has been unfairly favoured here in that this off-line procedure requires a certain amount and type of experimental data for the model identification of the process.

This controller is going to be used to compare the results obtained with the auto-tuned MPC.

3.3 MPC

For the implementation, is worth mentioning that it is advisable to use a a graphical language such as *ladder logic* or *function block* because technical staff are much more familiar with this than structured text; also it is easier to maintain and debug for the changing nature of bits, timers, counters etc. while being monitored. However, a significant barrier for MPC implementation is that operations between vectors and matrices are not defined in any of the supported IEC 1131.1 languages, thus, all of these operations have to be programmed from scratch.

The complete structure of the proposed MPC program (based on subsections 2.3–2.5) is shown in Fig. 3. The algorithm has been programmed in the High Priority Periodic Execution Group [4] called `AutotunedMPC` which contains the routines summarised in Table 1. The software design, matrix formations, sequence of execution and the computation of the calculated output is described in detail in (Valencia-Palomo and Rossiter, 2008); with the exception of the routine `Optimisation` which is new to this paper and developed to include constraint handling, as described in subsection 2.4.

It can be seen from the properties of the controller with the RSLogix programming tool (Fig. 4) that the programm uses 17% of the available storage of the PLC including memory requirements for I/O, running cache and other necessary subroutines.

Finally, the input parameters for the program are only those who are related with de model, i.e. dead time t_d,

[4] This periodicity is set up with the chosen sample time.

Table 1. Routines and programming languages.

	Routine name	Programming language
1	`MPC_Main`	Ladder logic
2	`Controller_Output`	Ladder logic
3	`Exp_Trajectory_Reference`	Ladder logic
4	`G_Matrix_Formation`	Structured text
5	`Matrix_Inverse`	Structured text
6	`Matrix_Multiply`	Structured text
7	`Matrix_Transpose`	Structured text
8	`Output_Predictions`	Ladder logic
9	`Plant_Simulation`	Ladder logic
10	`Optimisation`	Structured text

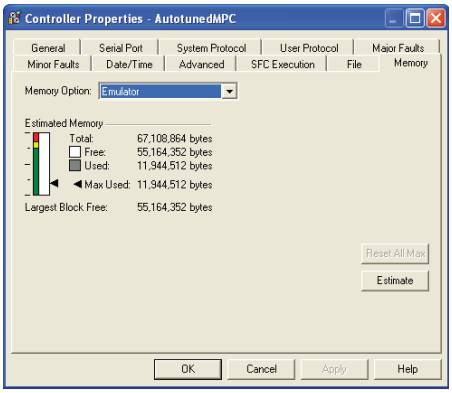


Fig. 4. MPC memory usage in the target PLC.

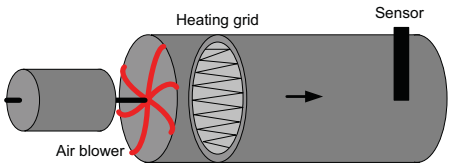


Fig. 5. Heating process.

rise time t_r, settling time t_s, overshoot M_P, gain K and sampling time Ts. The tuning is done online in the first scan of the program and is not repeated after, however, some mechanisms to update the model of the plant and tuning parameters can be embedded.

4. EXPERIMENTAL LABORATORY TESTS

This section shows the experimental results from applying the MPC law via a PLC on a first and a second order plant. For both processes the interest is tracking of step references, which is the most common situation in industry. The PID/MPC experiments ran under the same conditions, in so far as this can be guaranteed.

4.1 First order plant – Temperature control

The first experiment is a heating process consisting of a centrifugal blower, a heating grid, a tube and a temperature sensor, see Fig. 5. The objective is to control the temperature at the end of the tube by manipulating speed of the blower (the input voltage of the D.C. motor).

The model of the plant for the PID off-line tuning is done by a least square estimator assuming a first order plant.

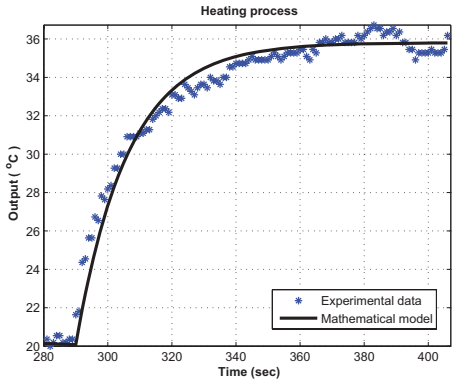


Fig. 6. Model validation of the heating process.

Table 2. Tuning parameters for the PID controller and input parameters for the auto-tuned MPC.

		Heating Proc.	**Speed Proc.**	
Cont.	Par.	Value	Value	Units
PID	P	1.590	0.325	–
	I	0.486	0.799	–
	D	0.0	0.0	–
MPC	Ts	10.0	0.50	Sec
	t_d	2.0	1.00	Samples
	t_r	38.0	4.90	Sec
	t_s	64.0	2.71	Sec
	K	16.0	190.0	–
	M_p	–	–	%

The resulting discrete model, with a sampling time of 10 sec. (which is also roughly the dead-time), is:

$$x(k+1) = 0.94x(k) + u(k)$$
$$y(k) = 0.94x(k)$$

The experimental validation of the mathematical model is shown in Fig. 6.

The tuning parameters for the PID controller and the input parameters for the Auto-tuned MPC are shown in Table 2. The second order approximate model for the predictions is built using standard analysis of the transient response of the plant i.e. nonparametric identification.

The simulation comparisons deploy a step change at $t = 50\,secs$ of the set point from 20 $^{\circ}C$ to 30 $^{\circ}C$; after the output reaches a new steady state, the process is disturbed at $t \approx 120\,secs$ by partially blocking the end of the tube. A new change in the set point to 20 $^{\circ}C$ is required at $t = 180\,secs$ without taking out the disturbance.

It can be seen in Fig. 7 that the plant output is successfully tracking the reference signal for both controllers. Of course this is a simple first order model and thus good control is to be expected. MPC has clearly given better control of overshoot and settling.

4.2 A second order plant – Speed control

This process consists of a motor fitted with a speed sensor, the control objective is to regulate the speed of the motor by manipulation of the input voltage. The same procedure as in the first experiment is applied. The mathematical model of the system with a sampling time of 0.5 sec is:

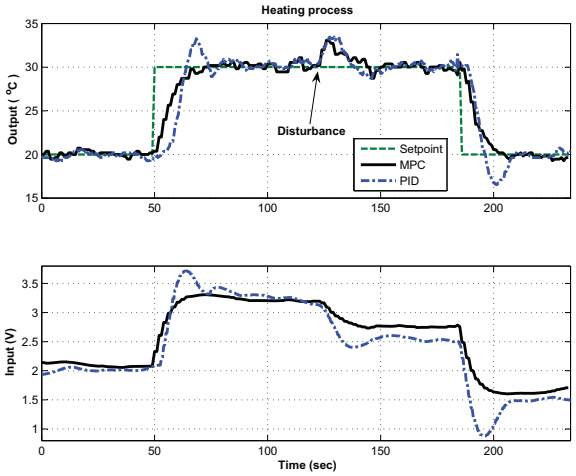


Fig. 7. Experimental test for the heating process.

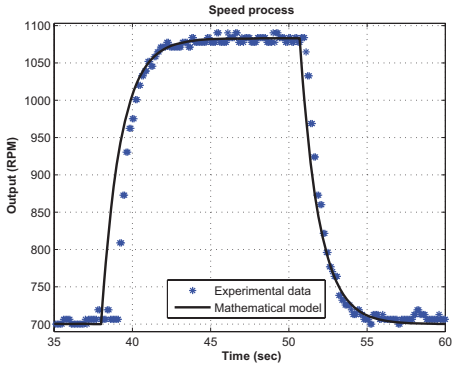


Fig. 8. Model validation of the speed process.

$$x(k+1) = \begin{bmatrix} 0.93 & -0.01 \\ 0.04752 & 0.9964 \end{bmatrix} x(k) + \begin{bmatrix} 1 \\ 0 \end{bmatrix} u(k)$$
$$y(k) = \begin{bmatrix} -0.01 & 3.71 \end{bmatrix} x(k)$$

The experimental validation is shown in Fig. 8 and the tuning parameters are in Table 2. Two set point step changes are demanded; once again, the results in Fig. 9 show that MPC and PID are tracking the set point accurately.

4.3 Performance indexes of the algorithms

The numerical performance indexes of the systems with the two different controller strategies are summarised in Table 3. Specifically the table shows the measures of performance are given by the cost function (J), the settling time (τ_s) and the overshoot (M_p). These numbers show that MPC performs similar to the standard PID controller but,in this case, with a much simpler auto-tuning procedure.

The constraint handling was not tested in a rigorous manner to let the PID controller act in the best possible scenario. Despite that, this simple control task finds its optimal value in saturation (Rojas and Goodwin, 2002).

To complete the analysis of the implemented program, the diagnostics tool from the hardware (shown in Fig. 10)

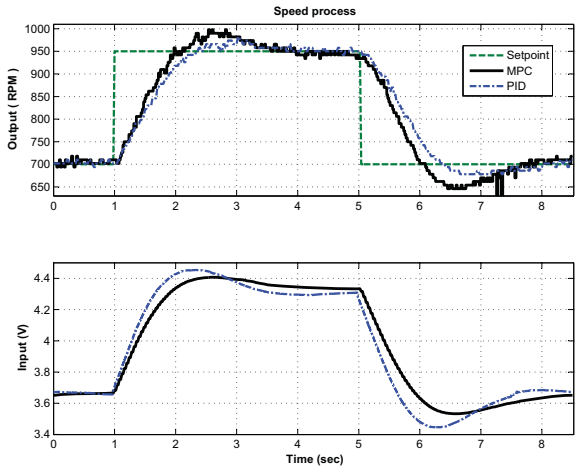


Fig. 9. Experimental test for the speed process.

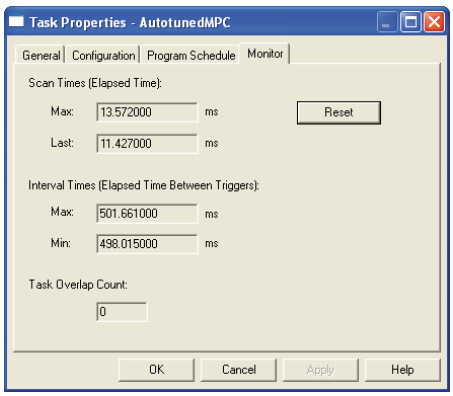


Fig. 10. Execution time and sampling jittering for the speed process.

Table 3. Performance indices for the systems.

	Heating process			Speed process		
	τ_s	M_p	J	τ_s	M_p	J
PI	31 sec	33.45 %	1645	3.3 s	16%	2,615
MPC	20 sec	0.0 %	1629	3.3 s	8%	2,816

displays that the time for scanning the program each sample time oscillates between 11.42 ms and 13.57 ms while the elapsed time between triggers (sampling instants) for the speed process oscillates between 498.01 ms and 501.66 ms. The significance of this is the potential to apply the algorithm on much faster processes.

5. CONCLUSIONS

This paper has made three contributions. First it has demonstrated that an MPC algorithm with systematic, albeit simplistic, constraint handling can be coded in an industrial standard PLC unit and with sample times of milliseconds. Secondly it has demonstrated that such an algorithm can make use of simplistic modelling information in conjunction with basic auto-tuning rules and still outperform an advanced auto-tuned PID whose design relied on far more information. Moreover the MPC includes constraint handling. Thus, thirdly the paper has demonstrated that MPC is a realistic industrial alternative to PID in loops primarily controlled with PLC units. This final contribution opens up the potential for much improved control of loops where PID may be a poor choice.

These results demonstrate the potential for implementing auto-tuned MPC within a PLC. Some issues the authors intend to pursue are: (i) demonstrate the algorithm in more challenging test rigs such as those with non-minimum phase behaviour and/or significant dead-times; (ii) consider extensions for unstable systems; and (iii) implement more advanced dual-mode type MPC algorithms and more advanced constraint handling facilities into the PLC.

REFERENCES

Camacho, E. and C. Bordons (2004). *Model predictive control*. 2nd ed.. Springer Verlag.

Clarke, D.W. (2002). Designing phase-locked loops for instrumentation applications. *Measurement* **32**, 205–227.

Clarke, D.W. (2006). PI auto-tuning during a single transient. IEE *Proceedings Control Theory and Applications* **153**(6), 671–683.

Froisy, J.B. (2006). Model predictive control — building a bridge between theory and practice. *Computer & Chemical Engineering*. **30**, 1426–1435.

Gyöngy, I.J. and D.W. Clarke (2006). On the automatic tuning and adaptation of PID controllers. *Control Engineering Practice* **14**, 149–163.

Maciejowski, J.M. (2002). *Predictive control with constraints*. Prentice Hall.

Morari, M., C. Jones and M. Zeilinger (2008). Low complexity MPC. In: *International Workshop on assessment and future directions in NMPC*. Pavia, Italy.

Richalet, J. (1993). *Practique de la commande predictive*. Hermes, France.

Richalet, J. (2007). Industrial application of predictive functional control. In: *Nonlinear Model Predictive Control, Software and Applications*. Loughborough, U.K.

Rojas, O.J. and G.C. Goodwin (2002). A simple anti-windup strategy for state constrained linear control. In: *IFAC World Congress*. Barcelona, Spain.

Rossiter, J.A. (2003). *Model predictive control: a practical approach*. CRC Press.

Rossiter, J.A. and P. Grieder (2005). Using interpolation to improve efficiency of multiparametric predictive control. *Automatica* **41**, 637–643.

Rossiter, J.A., B. Kouvaritakis and M.J. Price (1998). A numerically robust state-space approach to stable-predictive control strategies. *Automatica* **34**(1), 65–73.

Valencia-Palomo, G. and J.A. Rossiter (2008). The potencial of auto-tuned MPC based on minimal plant information. In: *Assessment and Future Directions of Nonlinear Model Predictive Control*. Pavia, Italy.

Valencia-Palomo, G., K.R. Hilton and J.A. Rossiter (2008). Predictive control implementation in a PLC using the IEC 1131.3 programming standard. *In review for the European Control Conference 2009*.

Zhu, Y., Z. Xu, J. Zhao, K. Han, J. Qian and W. Li (2008). Development and application of an integrated MPC technology. In: *Proceedings of the 17th IFAC Wolrd Congress*. Seoul, Korea.

The Effect of Tuning in Multiple-Model Adaptive Controllers: A Case Study

Ehsan Peymani* Alireza Fatehi**
Ali Khaki Sedigh***

Advanced Process Automation & Control (APAC) Research Group, K. N. Toosi University of Technology
Tehran, Iran.

(e-mail: *ehsan.peymani.f@ieee.org, **fatehi@kntu.ac.ir, ***sedigh@kntu.ac.ir)

Abstract: In this paper, two types of multiple-model adaptive controllers are practically evaluated on a laboratory-scale pH neutralization process. The first one is supervisory switching multiple-model adaptive controller (SMMAC) whose model bank is fixed and selected a priori, and another one is a controller based on multiple models, switching, and tuning strategy (MMST) which uses the possibility of model bank tuning. In addition to investigation of the effect of tuning, the advantage of a disturbance rejection supervisor is studied. Various experiments and exhaustive numerical analyses are provided to assess the abilities of the proposed algorithms.

Key Words: multiple models, adaptive control, pH control, switching control, pole-placement control

1. INTRODUCTION

Multiple-model adaptive control is a promising approach to control complex, nonlinear, and time-variant systems. On the grounds that a very complicated system is decomposed to simpler and smaller ones in this approach, and therefore, a large set of model uncertainty is converted to smaller sets, this approach results in a robust controller. It is, also, called an intelligent approach if intelligence is defined as rapid and appropriate response to large and sudden variations in a system (Narendra and Balakrishnan, 1997).

Multiple-model approaches are well-known not only in control but also in identification and estimation (Johansen and Murray-Smith, 1997). Multiple-modeling means that a set of models describes a dynamical system instead of one lone model. According to the type of contribution of members of this set to construct the global model of the process, switching or interacting multiple-model approaches are obtained. In switching multiple-model control strategies, at each instant, one model of the bank is selected as the global model, and the controller is designed according to the parameters of the model. This kind of control strategy has been evaluated in many subject areas such as robotics (Czlzz and Narendra, 1996), flight control (Boskovic and Mehra, 1999), aerospace applications (Karimi and Landau, 2000), and process control (Pishvaie and Shahrokhi, 2000; Gundala, Hoo, and Piovoso, 2000).

It is obvious that model bank significantly affects control performance. Thus, it is critical to have a model bank that considers all possible operating points. Since all possible operating points are not known a priori, increasing the number of model bank members may be a solution. However, in addition to intensifying computational burden, there is a chance of deteriorating performance owning to excessive competition of unnecessary members (Li and Bar-Shalom, 1996). Another solution is model bank tuning.

Model bank tuning means that beside each fixed models there is an adaptive model which starts to adjust itself from the location of the fixed model after the fixed model was chosen by the supervisor of the control system. Consequently, a quite new control strategy is introduced and called *multiple models, switching, and tuning* (MMST) (Narendra and Balakrishnan, 1997; Narendra and Xiang, 2000).

The main objective of this paper is to study how the possibility of tuning affects the effectiveness of a multiple-model adaptive controller. To accomplish the objective, we experiment two multiple-model controllers on a pH pilot-plant. The pilot-plant was designed and constructed by members of the research group at KNTU. Figure 1 shows this plant. The secondary objective is to investigate practically a disturbance rejection supervisor, introduced firstly in (Peymani et al, 2008). Since there is no buffer stream in the pilot-plant, the process keeps its high nonlinearity such that the static gain of the process can vary more than 70 times for the entire operating range.

The paper is organized as follows. After introduction, the principles of multiple-model adaptive controllers are reviewed. Two control strategies are presented: one has tuning possibility and the other does not. In the next section, section 3, a disturbance rejection supervisor is designed and its specifications are stated. Then, section 4 is allocated for the pH pilot plant description. Application results are, also, provided in this section. Finally, conclusions end the paper.

2. THE BASIS OF MULTIPLE MODELS ADAPTIVE CONTROLLERS

The control strategies utilized in this paper are based on supervisory switching multiple-model adaptive controllers. In these controllers, the whole nonlinear system is divided into several linear systems which can be represented more exactly by a set of simple linear models, called *model bank*.

Fig 1. pH pilot-plant at ACSL, K. N. Toosi University of Technology

That which model and when must be selected is the duty of the *supervisor*. The supervisor determines the appropriate model according to a *switching scheme*.

Let us presume that we have the simplest form of model bank which contains N fixed-parameter models. In fact, each model is a predictor anticipating the future output of the process in accordance with the last measured input and output. The difference between the output of the real plant and of each model is sent to the supervisor to calculate the identification performance index according to eq. (1):

$$J_s(t) = \alpha e_s^2(t) + \beta \sum_{k=1}^{M} \lambda^k e_s^2(t-k), \quad 0 < \lambda \le 1, \alpha, \beta, M > 0 \tag{1}$$

in which $e_s = y - \hat{y}_s$, and α, β, λ and M are the free-design parameters. Pole-placement control design method is utilized in this paper. It is a two-degree of freedom dynamical output feedback controller having the form of:

$$R(q)u(t) = T(q)u_r(t) - S(q)y(t) \tag{2}$$

where R, S, and T are calculated after the process model was selected. The process is viewed as a first order plus time delay model (FOPDT). To find the appropriate values for the controller parameters, it is necessary to define a model reference, a priori, regarding the control objective, and solve a polynomial Diophantine equation on-line. For more details about the controller design method see (Astrom and Wittenmark, 1995; Peymani et al, 2008).

The significance of model bank on the performance of the control system is beyond dispute. The more precise a model bank represents the plant, the better the control system performs. Difficulties arise from the problem of decomposing a plant into efficient smaller linear subsystems. Moreover, a real-world plant inevitably encounters variations which are able to make new operating conditions.

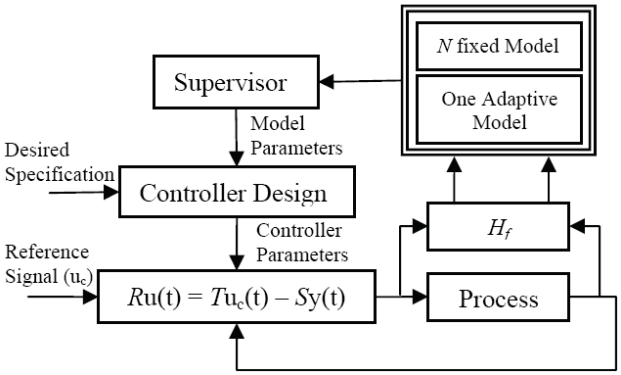


Fig 2. The block diagram of the multiple models, switching, and tuning adaptive controller

If an unpredicted one comes about, the control performance may become weak. To cover more states of the process, the number of members of the model bank should be increased. This solution is not suitable because a bank with too many members may deteriorate the performance in addition to excessive complexity burden (Li and Bar-Shalom, 1996).

Tuning of the current model is another solution to solve this problem. In this idea, after a fixed model is selected, an adaptive model starts to adjust itself to the process condition from the location of the fixed model. Thus, convergence of the adaptation algorithm may be reached soon, and the adaptive model may describe the process more precisely than the selected fixed model does. This model is called *re-initialized adaptive model*, and can be built by a recursive least-squares (RLS) identification method with exponentially forgetting factor (Astrom and Wittenmark, 1995). This control structure containing both fixed and adaptive models in the bank is also called *Multiple Models, Switching, and Tuning* (MMST) control strategy and firstly is introduced by Narendra et al. Thus, this adaptive control strategy possesses a two-stage identification unit: the switching stage to overcome sudden and large changes of the process and the tuning stage to track slow and gradual process variations. Figure 2 shows this control strategy.

Switching Scheme: Decision-making part in MMST is done by supervisor as follows. At each sampling time, performance indexes of N fixed models and one re-initialized adaptive model are updated. Then, in the switching stage, the best fixed model whose index is smaller than the product of other indexes by h_S is selected. The factor h_S is the hysteresis constant for the switching stage. After each change of the best fixed model, the adaptive model is reinitialized by the parameters of the fixed model unless the identification performance of the adaptive model is better than the best fixed model. The decision can be made by comparing their performance indexes.

After the switching stage, the tuning stage triggers. In this stage, the supervisor orchestrates which of the adaptive or the best fixed model is appropriate for control. This stage owns another hysteresis constant, represented by h_T. If the index of

the adaptive model is smaller than the multiplication of the index of the best fixed model by h_T, the adaptive model is used to design the controller. At this time, the process is controlled by an *adaptive pole-placement controller* (APPC). With the same logic, the supervisor determines whether it is appropriate to use the fixed model for control. As mentioned, SMMC is simpler than MMST control strategy and dose not have tuning stage and adaptive model, so switching stage is simpler.

3. DISTURBANCE REJECTION SUPERVISOR

All adaptive controllers need the process information to adapt itself to the current condition. The process has to be excited very well in order to collect appropriate information. In process control, the set-point rarely changes. Nonetheless, disturbances occur occasionally. Hence, disturbances can be considered as the staple source of excitation.

When a disturbance happens, at first the process output deviates from the reference input. Then, the controller reacts against this action. Consequently, it is possible to divide the process output into two parts. The first part is the duration which the disturbance, as an unmeasured input, affects the process output, and the second is the one that the control signal derives the system in spite of disturbance. Excitation resulted from the first part is adverse for identification, but excitation caused by the second part which the control signal dominantly affects the system is proper for process identification.

Then, discarding irrelevant data is vital to enhance on-line system identification. This aim is achieved if the time when a disturbance occurs is detected; that is, the identification stage is interrupted in the first part of excitation caused by a disturbance. An idea to discover disturbance occurrence is proposed in (Hagglund and Astrom, 2000) and is shown in Fig. 3, in which y_f and u_f are high-pass filtered process output and input, respectively. According to this idea which is proposed for positive-gain systems, when filtered process input and output are larger than predefined thresholds and have opposite signs, a disturbance has occurred. The time when the control action is appropriate for identification can be estimated based on the maximum value of the filtered process output.

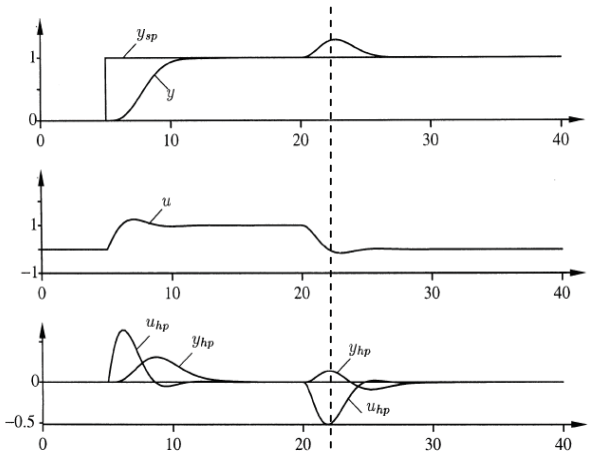


Fig 3. Response of a linear system in close-loop feedback on condition that a disturbance occurs (Hägglund and Astrom, 2000).

According to the extension of this supervisory function for multiple-model controllers, which is proposed by the authors in (Peymani et al, 2008), when a disturbance discovered, both switching and tuning stages are not allowed to work, and the controller is designed based on the best fixed model. After the first part of excitation, the tuning and switching parts are permitted again. In MMST strategy, it is possible to re-initialize the adaptive model to converge rapidly after negative excitation of disturbance.

4. APPLICATION RESULTS

The aim of this paper is to study the effect of tuning in the model bank of multiple-model adaptive control strategies by evaluating two multiple-model based controllers on a pH pilot-plant. After pH pilot plant description, application results are provided.

4.1 pH pilot-plant Description

This plant was designed and constructed in K. N. Toosi University of Technology (Fig. 1). In this process, as shown in Fig. 4, there are four streams: acid, base, water, and effluent. The acid stream is the process stream; its concentration and flow rate are presumably constant. The base stream is the titrating stream whose concentration is constant but flow rate is calculated by the controller to regulate pH of the effluent stream. The effluent stream has a constant flow rate. Moreover, this plant is composed of a continuous stirred tank reactor (CSTR) where chemical components are mixed, a pH sensor, a level sensor, and three dosing pumps which inject water, base, and acid with precise flow rate. To keep the level of the tank constant, a classical PI controller is designed which uses water stream flow rate as the control signal. Figure 5 displays the block diagram of this control system.

It is valuable to mention that aquatic solution of acid acetic (a weak acid) and sodium hydroxide (a strong base) are used as process stream and titrating reagent, respectively. However, there is no buffer stream in this pilot-plant. This feature makes this process highly nonlinear. Figure 6 depicts the titration curve of the process with typical concentrations. As it can be seen, it has a steep and large increase in its derivative. This curve is calculated for a batch process because calculating the titration curve of a continuous process is very difficult owning to inherent nonlinearity and various external disturbances. Figure 7 reveals that the static gain can vary more than 70 times for pH in the range of 5 to 10. It is worth mentioning that the measurement noise of the process varies with pH. It is about 5e-5 outside the range of [7, 8.2], whereas it is about 1e-3 inside the range.

4.2 Experimental Evaluation

In this section, two multiple models adaptive controllers, proposed earlier, are considered and their parameters are presented specifically. To have comparable situations, the parameters of both controllers are chosen the same. Indeed, the only difference between them is that MMST based controller has the possibility of model bank tuning, while supervisory switching multiple models adaptive controllers (SMMC) has a fixed-parameter model bank. Anyway, to construct a fixed model bank, the process is controlled by a classical adaptive pole-placement controller in varying pH values.

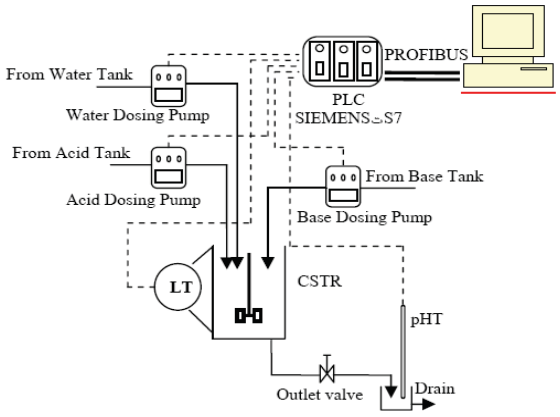


Fig 4. Schematic diagram of pH pilot-plant

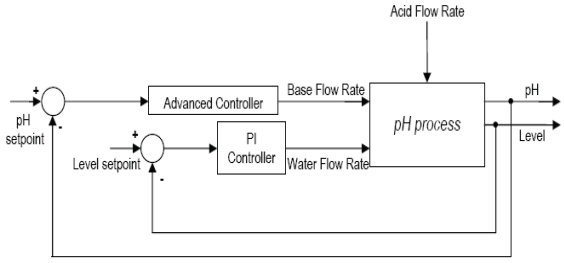


Fig 5. Block diagram of the control system

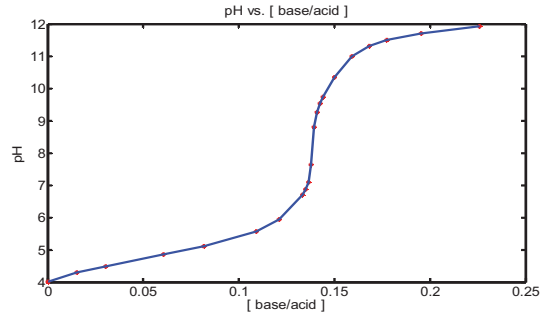


Fig 6. Titration curve of the process stream in the batch form.

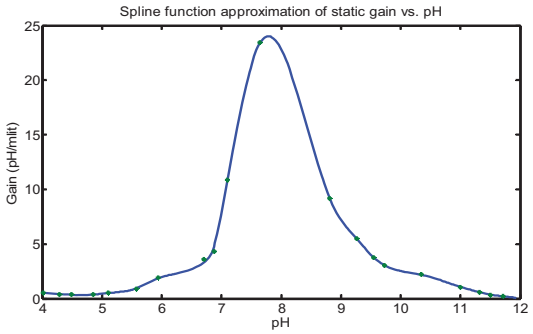


Fig 7. Variations of static gain of pH pilot-plant versus pH

After convergence of the adaptive model, the model is saved as the model of current operating condition. Table I collects models of various operating points. Before identifying, data are passed through a band-pass filter to discard bias and attenuate the adverse effect of noise. Hence, as shown in Fig. 2, filter H_f is located at the beginning of the identification loop, which is:

$$H_f(q) = \frac{(q-1)}{(q-\alpha)^2}(1-\alpha)^2 \tag{3}$$

The parameter α is 0.9652. Then, the model reference for pole-placement control design method is:

$$G_m(q) = \frac{0.0012}{(q-0.9652)^2} q^{-8} \tag{4}$$

The parameters of performance index are selected as $M = 50$, $a = 320$, $b = 100$, $\lambda = 0.985$. Hysteresis factor for SMMAC equals to 0.80, and switching and tuning hysteresis factors in MMST are chosen 0.85 and 0.75, respectively.

We consider a sequence for step-like set-point changes containing three kinds of variations: 1) small changes which means set-point varies from 5 to 10 one by one (Fig. 8.a); 2) medium changes which means set-pint changes from 6 to 8 and to 10, and then returns to 6 in the same manner (Fig. 8.b). These variations are known as the most difficult ones in this process; 3) large changes that contain some variations larger than two units in pH (Fig 8.c). Moreover, the acid feed rate is considered as a measured disturbance. Its nominal value is 30% of maximum power of the corresponding pump. The sequence of 30% → 18% → 42% → 30% is considered as disturbance sequence.

A criterion is defined in order to compare the results numerically, so we choose a two-part measure. Each part is calculated individually for each change in set-point or each disturbance. It is:

$$\begin{aligned} C &= \sum MSWE \; + 10 \sum O.C. \\ MSWE &= \frac{100}{n-1} \sum \left((y(t) - y_r(t))^2 e^{0.005\,t} \right) \\ O.C. &= 1 - \left| \frac{\sum (y_f)}{\sum |y_f|} \right| \end{aligned} \tag{5}$$

in which *O.C.* is a measure of oscillation. Mean square weighted error (*MSWE*) shows how much the process output is similar to y_r, the model reference output or desired output. It weights the errors exponentially in time.

At first, our aim is to regulate pH of effluent process at different values. Thus, we use the same set-point sequence with various changes to assess how much each controller is able to drive the process similar to the desired output. Fig. 8 shows the results, and table II gathers numerical analysis.

The second test is to evaluate the ability of the controllers to reject external disturbances. Figures 9 to 11 show the results of this test. The effect of disturbance rejection supervisor is demonstrated in Fig. 10 and Fig. 11. Table III compares the disturbance rejection abilities numerically.

4.3 Discussion

This section is allocated for application results. According to Fig. 8, the presence of tuning in the model bank of a multiple-model adaptive control can improve transient response. In this figure, MMST control algorithm has a smoother response, especially for pH around 7.5. Changing from 6 to 8 (and vice versa) is the most difficult change in this plant because the control system has to drive the process from a low-gain operating point to the highest one. The same result can be derived from disturbance rejection part. Hence, it is evident that MMST strategy has a better performance than the other strategy without tuning possibility does.

Figures 9, 10, and 11, and table III demonstrate that the disturbance rejection supervisor can help the supervisor to make a better decision, so this additional supervisory function is satisfactory. Generally, according to Fig. 11, the disturbance rejection supervisor is the reason to a faster rejection.

Table 1 Local models of the process, time delay neglected to be shown equals to 9 sampling times (45 seconds).

Model no.	1	2	3	4
Local Model	$\frac{0.0070}{z-0.985}$	$\frac{0.0166}{z-0.990}$	$\frac{0.0044}{z-0.970}$	$\frac{0.001}{z-0.970}$

5. CONCLUSION

The proposed MMST control strategy is modified slightly from the original version and the switching stage is transparently separated from the tuning phase. Moreover, to constrain the speed of switching, hysteresis constants are used. The presence of tuning in multiple-model adaptive controllers has positive effect in performance. The application results, which are provided for both tracking and disturbance rejection problems by implementation two multiple-model controllers on a pH pilot-plant, prove that the response of MMST is smoother and more similar to desired output. Also, they uncover that the possibility of tuning in identification loop prevents the model bank to be specified to a certain process. In fact, tuning tries to generalize the model bank. Furthermore, the application results reveal that the presence of the disturbance rejection supervisor can augment the effectiveness of multiple-model adaptive controllers to face load disturbances without imposing significant computational burden. The disturbance rejection supervisor is designed for positive-gain systems.

It can be claimed on condition that the model bank has a severe deficiency to describe the process, SMMC may result in instability, while MMST can stabilize the process because of the presence of tuning.

Table 2 Comparison between SMMC and MMST in tracking

	Small	Medium	Large	Overall
SMMC	32.37	52.61	63.29	148.27
MMST	27.81	25.79	63.50	117.10

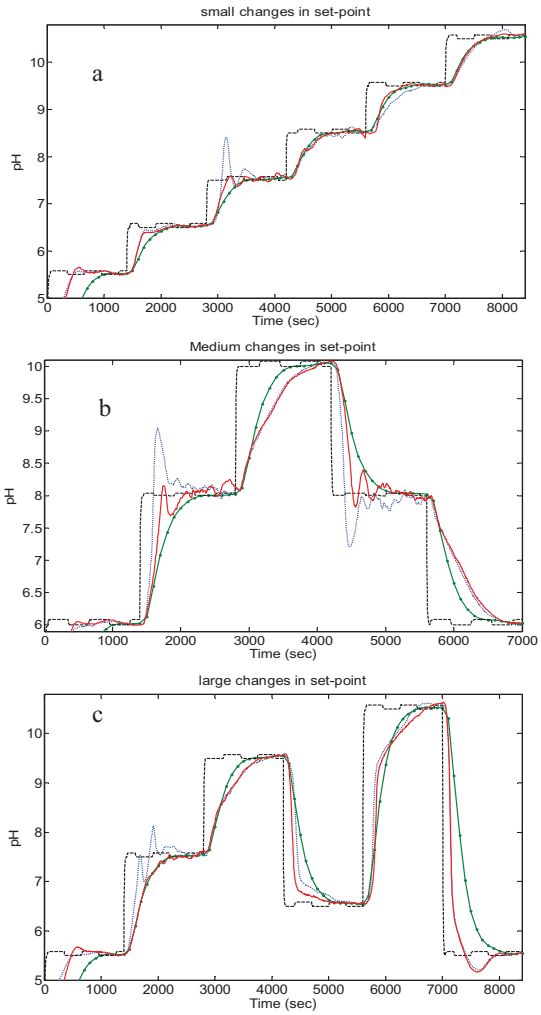


Fig 8. Evaluation of the effect of model bank tuning in tracking problem; a) small, b) medium, c) large changes in set-point; solid line (MMST), dash line (SMMC), and dash-dot line (desired output)

Table 3 Comparison of disturbance rejection ability between SMMC and MMST.

Disturbance Rejection	Disturbance Rejection Supervisor	
	OFF	ON
SMMC	167.38	90.31
MMST	75.07	66.31

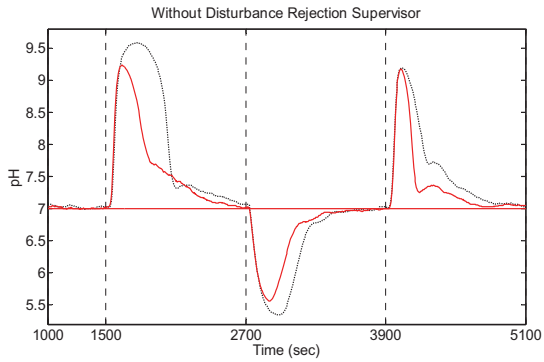


Fig 9. Disturbance rejection of both controllers without disturbance rejection supervisor; solid line (MMST) and dotted line (SMMC)

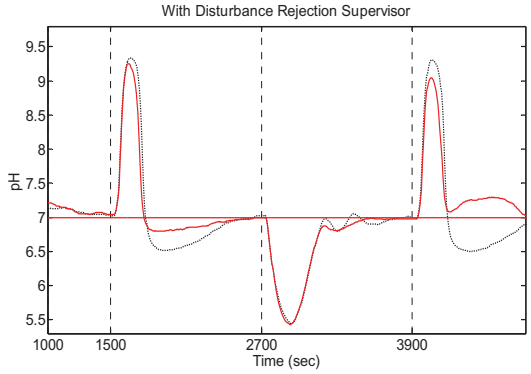


Fig 10. The effect of disturbance rejection supervisor; solid line (MMST) and dash line (SMMC)

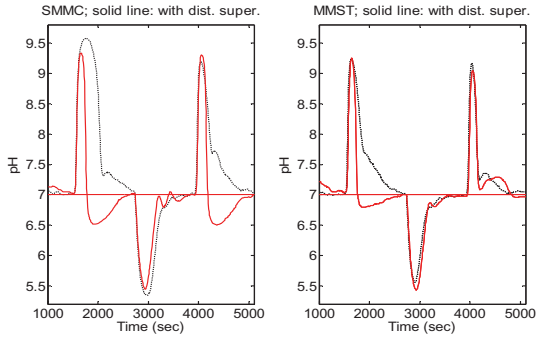


Fig 11. The effect of disturbance rejection supervisor, comparison in each controllers, dash line (without the supervisor) and solid line (with the supervisor); left: SMMC and right: MMST

REFERENCES

Astrom, K. J. and Wittenmark, B. (1995). *Adaptive Control*, 2nd ed.. Addison-Welsey, NY.

Boskovic, J. D., and Mehra, R. K. (1999). Stable multiple model adaptive flight control for accommodation of a large class of control effector failures. In Proceedings of the 1999 American control conference (Vol. 6, pp. 1920-1924), San Diego, CA, June 1999.

Czlzz, M. Kemal and Narendra, K.S., (1996). Intelligent Control of Robotic Manipulators: A Multiple Model Based Approach. The International Journal of Robotics Research, 15(6), 592-610.

Gundala, R., Hoo, K.A., and Piovoso, M.J., (2000). Multiple model adaptive control design for a MIMO chemical reactor. Ind. Eng. Chem. Res., 39(6), 1554-1564.

Hägglund, T. and Astrom, K. J., (2000). Supervision of adaptive algorithms. Automatica, 36(8), 1171-1180.

Johansen, T. A. and R. Murray-Smith (1997). The operating regime approach. In: Multiple Model Approaches to Modelling and Control. (R. Murray-Smith and T. A. Johansen, Eds.). Taylor & Francis Inc.. Bristol, PA. 3-72.

Karimi, A. and Landau, I. D., (2000). Robust Adaptive control of a flexible transmission system using multiple models. IEEE Transactions on Control Systems Technology, 8(2):321-331.

Li, X.R., and Bar-Shalom, Y., (1996). Multiple-model estimation with variable structure. IEEE Transactions on Automatic Control, vol. 41, No. 4, pp. 478-493.

Narendra, K.S. and Balakrishnan, J., (1997). Adaptive Control Using Multiple Models. IEEE Trans. on Automatic Control, 42(2), 171-187.

Narendra, K. S. and Xiang, C., (2000). Adaptive control of discrete-time systems using multiple models. IEEE Trans. on Automatic Control, 45(9), 1669-1686.

Pishvaie, M. R. and Shahrokhi, M., (2000). pH Control Using the Nonlinear Multiple Models, Switching, and Tuning Approach. Ind.& Eng. Chem. Res., 39(5):1311-1319.

Peymani, E., Fatehi, and A., Khaki-Sedigh, A., (2008) A disturbance rejection supervisor in multiple-model based control, proc. of International conference on control, IET, Manchester, UK.

Slug-flow Control in Submarine Oil-risers using SMC Strategies ⋆

Pagano, D. J. * **Plucenio, A.** * **Traple, A.** *

* *Departamento de Automação e Sistemas, Universidade Federal de Santa Catarina, 88040-900 Florianópolis-SC, Brazil*
e-mail: {daniel, plucenio, traple}@das.ufsc.br

Abstract: In this paper we propose different Sliding Mode Control (SMC) strategies to control slug-flow oscillations in submarine oil-risers. The main idea is to design a switching control law that induces a sliding bifurcation on the system, changing its dynamics and, in this way, controlling the amplitude of a limit cycle. Simulation results were obtained using OLGA Scandpower software in order to compare the different SMC strategies.*Copyright©2009 IFAC*

Keywords: Production oil system, Submarine oil-riser, Slug-flow control, Non-smooth dynamical systems, Sliding Mode Control, washout filter

1. INTRODUCTION

Transportation of multiphase fluid (oil, gas and water) is an important task in the oil industry. Nowadays, there is a trend to increase the number of satellite wells and the length of risers between clusters of wells and off-shore production systems. Besides, the increasing depths of oil wells produces several new multiphase transport problems, see Storkaas and Skogestad (2004), Storkaas (2005). In this scenario a common problem is the phenomena so called slug-flow characterized by the intermittent axial distribution of gas and liquid. The pressure and flow rate oscillations induced by the slug-flow can provoke several undesired effects on the surface equipments. These types of disturbances can cause serious problems in the input of the multiphase flow separator, deteriorating the separation quality and causing level overflow (Godhavn et al. (2005)). In short, the slug-flow phenomena in submarine risers cause several problems to the oil off-shore industry. The suppression of this type of oscillations by means of feedback automatic control methods can be applied to stabilize the flow in risers and, consequently, minimize the problems on the separator. At the same time, two other benefits can be obtained: (i) in cases where the oil is pumped from sea bottom, energy consumption is minimized; (ii) in cases of risers connected to wells with natural or artificial lift flows, higher production is obtained by minimizing the pressure in front of the well perforated zones.

A schematic diagram of a riser used in an oil production off-shore system is shown in Fig. 1 with parameters shown in Table 1. This system was simulated in OLGA [1] .

In Fig. 1, bottom and top riser pressures P_1 and P_2, respectively, are measured in [Pa] units and the control action is applied on the production choke. Modelling this system is quit complex since it involves partial differential equations. A simplified third order dynamical model developed in ordinary differential equations can be found in Storkaas and Skogestad (2004), Storkaas (2005). The bifurcation diagram considering the

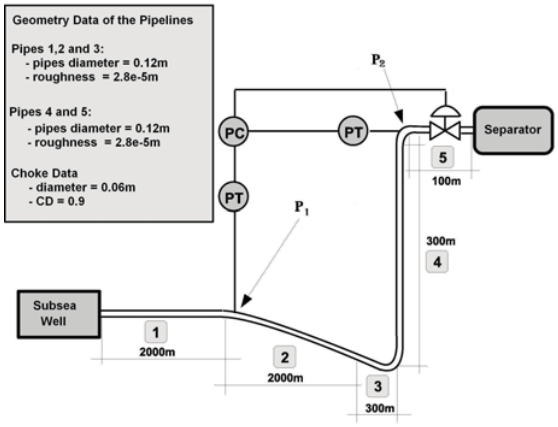


Fig. 1. Oil-riser system set-up simulated in OLGA.

Table 1. Parameter values of the riser setup.

Parameter	value	unit
Mass flow rate entering the riser	5	$Kg.s^{-1}$
Separator pressure	5.10^6	Pa
Gas void fraction	5	%
Temperature in the riser output	22	^oC
Temperatura in the well	62	^oC

choke opening as a bifurcation parameter (see Fig. 2), was obtained based on OLGA data simulations for a mass flow rate entering the riser equal to $5Kg.s^{-1}$ and a pressure separator of 5.10^6Pa. The bifurcation diagram of Fig. 2 is qualitatively similar to the diagram shown in Storkaas and Skogestad (2004). The stable and unstable equilibria manifold are depicted in Fig. 3. In this figure we show also the curves corresponding to maximal and minimum values of the limit cycle. A projection

⋆ Partially supported by *Agencia Nacional do Petroleo, Gas Natural e Bio-combustiveis* under project PRH34-ANP/MCT. Daniel J. Pagano was partially supported by grant PQ-310281/2006-7 from CNPq - National Council for Scientific and Technological Development/Brazil.

[1] Multiphase flow software simulation commercialized from Scandpower.

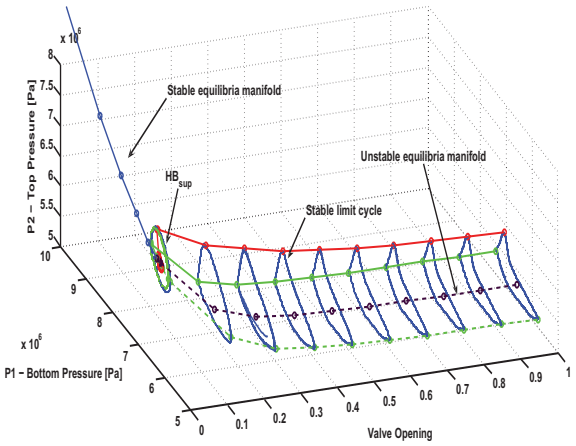


Fig. 2. Bifurcation diagram considering the choke opening as the bifurcation parameter. A stable limit cycle undergoes from a supercritical Hopf Bifurcation (HB_{sup}).

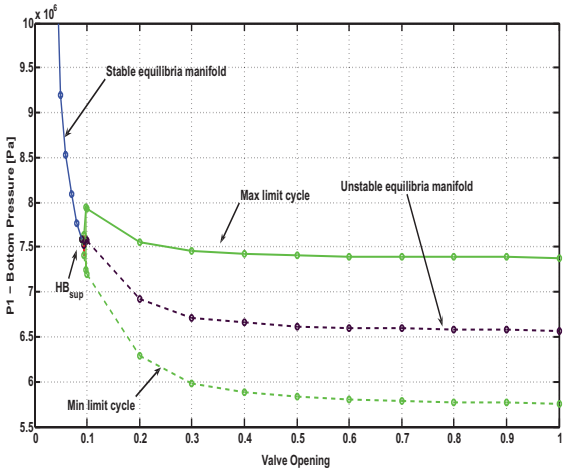


Fig. 3. Bifurcation diagram in $(u(t), P_1)$ plane.

of the limit cycles for different choke openings in the (P_1, P_2)-plane is shown in Fig. 4. As we can observe in this picture, the relation between P_1 and P_2 pressures on the stable (unstable) equilibria manifold can be approximated by a straight line. As can be seen in Figs. 2-4, a supercritical Hopf Bifurcation takes place, at the point HB_{sup} in the diagram, giving rise to a stable limit cycle. Thus, without active feedback control it is necessary to operate the system with choke opening below 10% in order to avoid output system oscillations. The pressure drop around the choke rises for low choke opening and this pressure drop is added to riser's bottom. High pressure for the same mass flow rate means higher energy consumption for sea floor pump applications. On the other hand, risers connected to natural or artificial lift wells may affect the pressure in front of the perforated zones leading to less oil production flow rate. Whatever the case it is desirable to have a steady flow with minimum pressure drop in the surface choke.

Several linear control laws to prevent slug-flow oscillations in submarine oil-risers have been proposed in different works, see

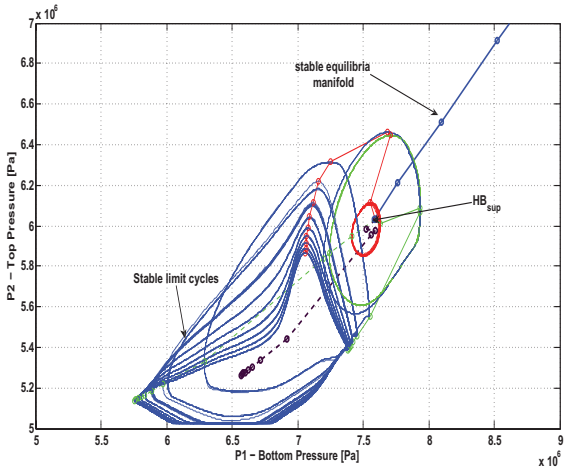


Fig. 4. Bifurcation diagram in (P_1, P_2) plane.

for instance Storkaas (2005), Godhavn et al. (2005). Linear controllers are only local solutions for this complex non-linear control problem. In this paper, as an alternative solution, we propose different non-linear control systems based on the Sliding Mode Control (SMC) theory.

The paper is organized as follows. In Section 2, the Proportional-Integral (PI) control law is revisited showing that it is not robust under disturbances in the input riser flow rate. In Sections 3 and 4, we propose different SMC strategies to control slug-flow oscillations. Our slug SMC washout strategy is presented in Section 5. Finally we discuss some of the limitations of these switching strategies and propose future improvements.

2. REVISITING THE PI CONTROL STRATEGY TO SUPPRESS SLUG-FLOW

In this Section we show by means of simulation results that the PI control law is not robust to disturbances in the input riser flow rate. A simulation test was made to evaluate the efficiency of the PI control. The PI control law is given by

$$u(t) = k_c[e(t) + \frac{1}{T_i}\int_0^t e(\tau)d\tau],$$

where $k_c = -7.92 \cdot 10^{-6} Pa^{-1}$, $T_i = 49.5s$, $e(t)$ is the error and the process variable is the pressure P_1. The PI discrete form implemented is given by

$$u(k) = u(k-1) + s_0 e(k) + s_1 e(k-1)$$

where $s_0 = k_c(1 + \frac{T_s}{T_i})$, $s_1 = -k_c$, $T_s = 1s$ is the sampling time and T_i is the integral time. PI control tuning was made using simple rules of adjusting since no mathematical model of reduced order for control design was available. The simulation setup was defined as:

(1) the choke opening is fixed at 20% and the corresponding operating point calculated from the equilibria manifold curve is $(P_1^*, P_2^*) = (6.93 \cdot 10^6[Pa], 5.56 \cdot 10^6[Pa])$;
(2) at $5000s$ the control is switched ON;
(3) a disturbance in the input riser flow rate is applied at $15000s$;
(4) the control is switched OFF at $25000s$.

Two flow rate disturbances were defined (i) from $5Kg.s^{-1}$ to $3.5Kg.s^{-1}$ and (ii) from $5Kg.s^{-1}$ to $3Kg.s^{-1}$. We use the

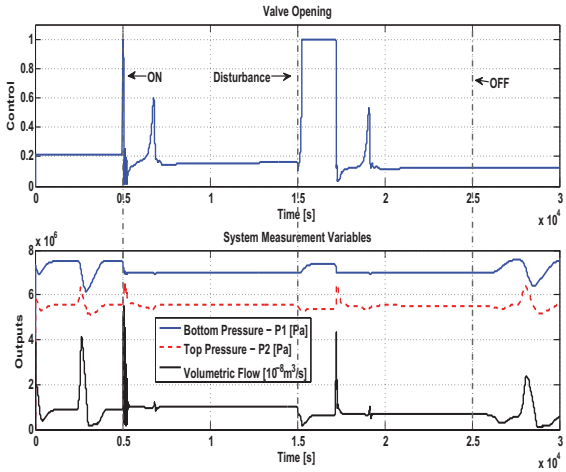


Fig. 5. System time response under *PI* control for a flow rate disturbance from $5Kg.s^{-1}$ to $3.5Kg.s^{-1}$.

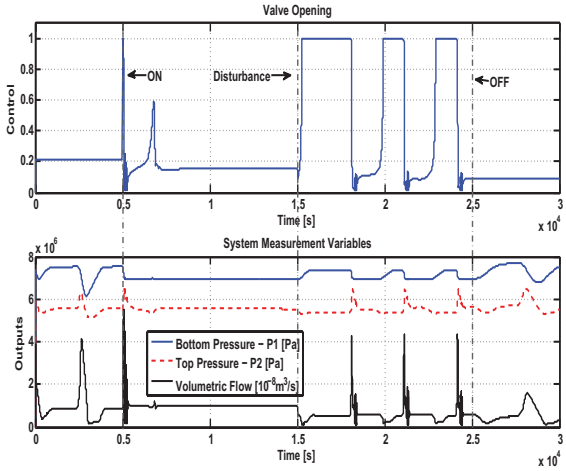


Fig. 6. System time response under *PI* control for a flow rate disturbance from $5Kg.s^{-1}$ to $3Kg.s^{-1}$.

previous simulation setup in order to obtain comparative results between the different slug control strategies.

Simulation results using the PI control are shown in Fig.5 for the first disturbance and in Fig.6 corresponding to the second disturbance. As can be seen, the PI control reject the first perturbation but it is not robust for the second disturbance.

In order to tackle this problem for large flow rate disturbances, three different Sliding Mode Control (SMC) strategies are proposed in the following Sections.

3. SLUG SMC STRATEGY

The main idea is to design a Sliding Mode Control (SMC) law (switching system) that induces a grazing-sliding bifurcation (see Angulo et al. (2005a)) on the system, changing its dynamics and, in this way, the amplitude of the target limit cycle is controlled. This type of non-smooth bifurcation introduces partial sliding motion along a sliding surface, reducing or suppressing the amplitude of the undesired limit cycle. In order to explain these ideas, consider a general system defined by

$$\dot{x} = F(x, u(x)) \tag{1}$$

where $x \in \mathbb{R}^n$ is the state vector of dimension n, and $u(x) \in \mathbb{R}$ is the control signal. The function $F(x) = (F_1, F_2, ..., F_n) : \mathbb{R}^n \rightarrow \mathbb{R}^n$, represents a non-smooth continuous system. We also assume that as a result of a Hopf bifurcation (continuous or not, see di Bernardo et al. (2008)), the system exhibits a steady state oscillatory behavior, where a stable limit cycle is the solution from (1).

The grazing-sliding bifurcation to suppress a limit cycle occurs when the limit cycle is crossed by a sliding surface that generates a grazing-sliding non-smooth transition where part of the trajectory of the limit cycle stands on the sliding surface as shown in Fig. 7.

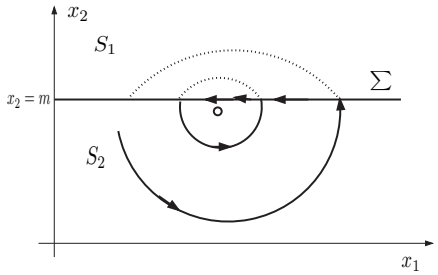


Fig. 7. Grazing-sliding bifurcation induced in the system.

For example, on a system with dimension 2, we consider a region S_1 of the form

$$S_1 := \{x = (x_1, x_2) : x_2 > m\}$$

for arbitrary m, being

$$\Sigma := \sigma(x) = \{x = (x_1, x_2) : x_2 = m\}$$

and

$$S_2 := \{x = (x_1, x_2) : x_2 < m\}.$$

With the variation of m, a grazing-sliding bifurcation occurs and the amplitude of the limit cycle is reduced or even eliminated.

Thus, the sliding mode control suggested is

$$u = u_0 + \Delta u \ \text{sgn}(\sigma(x)) \tag{2}$$

where $\sigma(x) = 0$ is the sliding surface, a function of the system's states that allow the changing of its dynamics; u_0 is the value of the control variable at the operating point and Δu is the maximum value that the control variable can assume from u_0.

The function $sgn(\cdot)$ can be defined as

$$\text{sgn}(\sigma(x)) = \begin{cases} -1, \text{ if } \sigma(x) < 0; \\ 1, \text{ if } \sigma(x) > 0. \end{cases} \tag{3}$$

or

$$\text{sgn}(\sigma(x)) = \begin{cases} 0, \text{ if } \sigma(x) < 0; \\ 1, \text{ if } \sigma(x) > 0. \end{cases} \tag{4}$$

Applying the above equations, we propose the following control law given by

$$u = u_0 + \Delta u \ \text{sgn}(\sigma), \tag{5}$$

$$\sigma(P_1, P_2) = P_2 - P_1 + \beta, \tag{6}$$

where $\beta = P_1^* - P_2^*$; $\Delta u = u_0 - u_{min}$; u_0 is the desired choke opening and u_{min} is the control value at the Hopf Bifurcation point. The switching surface is defined as $P_2 = P_1 - \beta$ and we define the $sgn(\cdot)$ to close the choke whenever $\sigma > 0$. The choke opening bias u_0 is defined at the riser desired operating point. At this point P_1^*, P_2^* are defined on the equilibria manifold curve, for a given mass flow rate of the riser input, as shown in Fig. 3. Choosing the surface choke opening bias u_o has to consider two factors. For one the value should be high enough in order to ensure a minimum pressure drop around the choke. On the other hand the bias should not be too far from values which can cause high pressure drops in order to answer quickly to disturbances. Choke opening close to 100% cause minimum pressure drop but depending on the choke characteristics a significant choke pressure drop can only be obtained for values smaller than 10%.

The control strategy can be interpreted as a mechanism to force an hypothetical steady flow rate which would be obtained without the slug flow behavior. For a constant input gas and liquid mass flow rate the pressure P_1 could be expressed as $P_1 = P_2 + \beta$ where β would take into account the gravity and friction terms of a pseudo stable flow. For instance, the simplest model is the homogeneous model given by

$$P_1 - P_2 = \frac{m_{gr} + m_{lr}}{A} g + \frac{f \bar{\rho} \bar{v}^2}{2 d_r} h, \tag{7}$$

where A is the section of the pipe; m_{gr} and m_{lr} are the mass of gas and mass of liquid in the riser; $\bar{\rho}$ is the mean density of the flow; $\bar{v}$ is the mean velocity of the flow in the riser; h is the height of the riser; f is the friction function; d_r is the diameter of the pipeline. In (7), first term corresponds to the gravity term and the second is the friction term.

Anytime the relationship is violated action is taken in the choke opening to force the desired P_1, P_2 relationship. Obviously this is done in a way that provides a desired choke opening which minimizes P_1 and consequently the energy used to lift the gas and liquid flow-rates entering the riser.

Time open-loop system responses are shown in Fig. 8. At $t = 5000s$ the proposed control system is turned on. At $t = 15000s$, a disturbance in the flow rate (from $5Kg.s^{-1}$ to $3.5Kg.s^{-1}$) was applied. It can be observed in Fig. 8 and Fig. 9 where the amplitude of the oscillations are decreased around the operating point when the control is switched on. As can be seen in Fig. 8, after the control is switched off, at $t = 250000s$, the oscillations back to the system. State space diagram, in (P_1, P_2)-plane, is depicted in Fig. 9.

The proposed SMC works well for small input riser flow rate disturbances but the control action switches permanently to maintain the equilibrium at the operating point.

4. A MODIFIED SLUG SMC STRATEGY

The SMC strategy development in Section 3 is not efficient to suppress pressure or flow oscillations in the riser since the control action switches permanently to maintain the equilibrium at the operating point. This would be very detrimental to the choke integrity. Another desired control characteristic is to be able to suppress the oscillations while keeping the choke nearly 100% opened. This represents significant less power needed to pump the multiphase fluid to the surface. In this Section, we propose a change in the control algorithm to minimize the switching in the control signal. The idea is to combine two control laws (i)

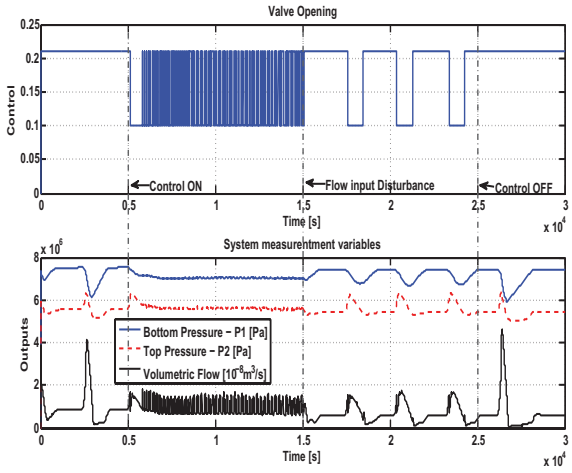


Fig. 8. Time system response (open-loop and with feedback control) with the SMC control strategy. a) choke opening; b) states of the system.

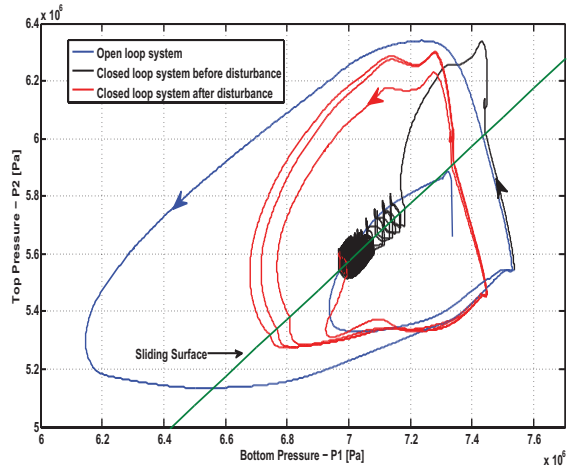


Fig. 9. State space diagram in (P_1, P_2)-plane

the control used in Section 3 and (ii) a discrete form of the PI control law as used in Section 2 by means of a convex function as

$$u(t) = \mu \; u_{SMC} + (1 - \mu) \; u_{PI}, \tag{8}$$

$$u_{SMC} = u_0 + \Delta u \; sgn(\sigma), \tag{9}$$

where

$$\sigma = P_2 - P_1 + \beta.$$

$$u_{PI}(k) = u_{PI}(k-1) + s_0 \; e(k) + s_1 \; e(k-1), \tag{10}$$

where u_{SMC} is the switching control law given by (9) and u_{PI} is the PI control (10), with $e(k) = P_1^*(k) - P_1(k)$.

The parameter $\mu = \mu(P_1, P_2)$ provides a smooth transition between the two control laws in such a way that if the trajectories are far away the equilibrium point then μ is close 1; otherwise μ is close to 0. It is defined as

$$\mu = \frac{1}{1 + e^{\gamma(\lambda - \delta)}}$$

$$\lambda(P_1, P_2) = \left(\frac{P_1}{P_1^*} - 1\right)^2 + \left(\frac{P_2}{P_2^*} - 1\right)^2 \tag{11}$$

where P_1^* is the operating point for bottom pressure and P_2^* is the desired value for the input choke pressure. Parameter values of (9), (10) and (11) are given in Table 2. Parameter

Table 2. Control law parameters.

Parameter	value	unit
u_0	0.2	
Δu	0.12	
s_0	$-8.08 \cdot 10^{-6}$	Pa^{-1}
s_1	$7.92 \cdot 10^{-6}$	$Pa{-}1$
γ	$8/\delta$	
δ	0.008	

β is defined as $\beta = P_1^* - P_2^*$. The system response with the proposed control law to a disturbance of the well mass flow rate is shown in Fig. 10 and the space state diagram is depicted in Fig. 11. The sample time was chosen as $T_a = 1s$. At

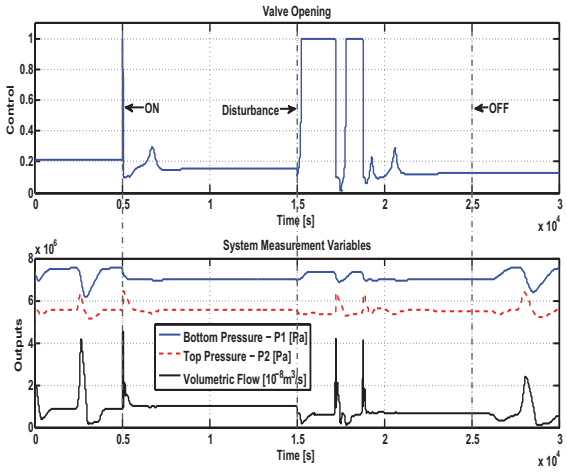


Fig. 10. Control and output system responses with the modified control law (8) for a disturbance input.

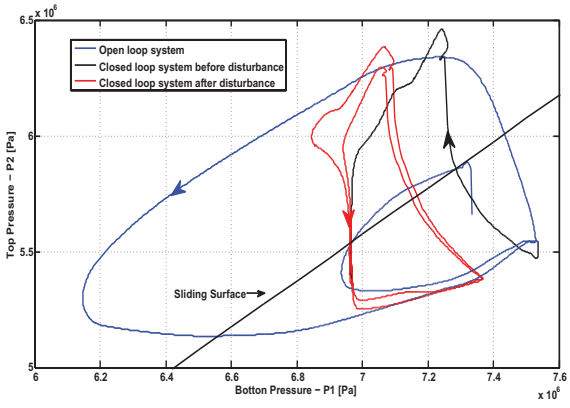


Fig. 11. State-space diagram of the system with the modified control law (8).

$t = 15000s$, the mass flow rate coming from the well is reduced from $5kg/s$ to $3.5kg/s$. This disturbance changes the operating process conditions. The proposed SMC control strategy control the system reduce the amplitude of the oscillations.

At $t = 25000s$, the control is turned off and the system starts to exhibit pressure and flow signal oscillations again. As can be seen the oscillations are back since the choke opening value is now in the instability region.

A disadvantage using the SMC algorithm discussed in this Section is that it is not possible to stabilize the system for large flow rate disturbances.

5. SLUG WASHOUT SMC CONTROL

All approaches presented so far for the slug control have used set-points to derive the control law. These strategies have a problem when there are changes in the fluid mass flow rate entering the riser. Even a stabilized flow rate will exhibit a different value both for P_1 and P_2 since higher mass flow rate will result in higher gravity and friction terms on the riser pressure drop as well as higher pressure drop in the surface choke. For the sliding mode control keeping the set-points for changes in the input mass flow rate means to request the system to operate in a limit cycle not sufficiently collapsed or to ask for an infeasible stabilized flow.

Since the practical objetive is to stabilize the flow keeping the surface choke with a minimum pressure drop, the idea of pressure set-point looses significance. One could say that the control problem is well solved if the pressures and flow rates do not oscillate while the surface choke is kept opened well above the opening which characterizes the beginning of the limit cycle. The idea is to develop a control strategy which supres the oscillation while keeping the choke opening operating around a desired opening value. If the oscillations are suppressed the resultant pressures will be a consequence of the input mass flow rate, fluid characteristics and the system geometry.

In order to attend the former constraints, we propose, in this Section, a new SMC strategy to reject mass flow rate input riser perturbations based on washout filters. Washout filters are intensively used to control chaotic systems by means of techniques based on bifurcation theory Wang and Abed (1995) and in flight control systems Lee and Abed (1991). Recently, washout filters were applied to power electronic converters in conjunction with SMC controllers in order to reject load disturbances Cunha and Pagano (2002). A washout filter is a high-pass linear filter that washes out steady-state inputs while passing transient inputs. The use of washout filters ensures that all the equilibrium points of the original system are preserved in the controlled system, i.e., their location remains unchanged.

The transfer function of a typical washout filter is given by

$$G_F(s) = \frac{s}{s+w} = 1 - \frac{w}{s+w},$$

where w denotes the reciprocal of the filter time constant which is positive for stable filter. We assume that it is possible to filter the inductor current x to achieve a new signal x_F and define an auxiliary variable z so that it is satisfied the output equation

$$x_F = x - z.$$

Then the effect of the washout filter can be represented by means of an additional differential equation, namely

$$\frac{dz}{dt} = w(x - z). \tag{12}$$

In our problem, we use two washout filters in order to filter the signals P_1 and P_2 in such a way that

$$\dot{z}_1 = w_1(P_1 - z) = w_1\tilde{p}_1$$

$$\dot{z}_2 = w_2(P_2 - z) = w_2\tilde{P}_2$$

where $\tilde{p}_1$, $\tilde{p}_2$ are the bottom and top filtered pressures, $w_1 = \frac{2\pi}{5}f_1$ and $w_2 = \frac{2\pi}{5}f_2$ are washout filter constants designed from the oscillatory frequencies f_1, f_2 measurement from the OLGA data simulation.

$$u_{WSMC} = u_0 + \Delta u \ sgn(\sigma), \tag{13}$$

where

$$\sigma(\tilde{P}_1, \tilde{P}_2) = \tilde{P}_2 - \tilde{P}_1. \tag{14}$$

Note that (14) is similar to (6) but now the parameter β is equal to zero. The sliding surface is now defined as $\tilde{P}_2 = -\tilde{P}_1$ and it does not depend on the operating point.

At $t = 10000s$ automatic control is turned on and at $30000s$ a well flow rate is reduced from $5kg/s$ to $3kg/s$.

At $50000s$ the control is again turned off and the system back to the oscillatory behavior. Simulation results are shown in Fig. 12. The state-space diagram in $(P_1 - P_2)$-plane is shown in Fig.

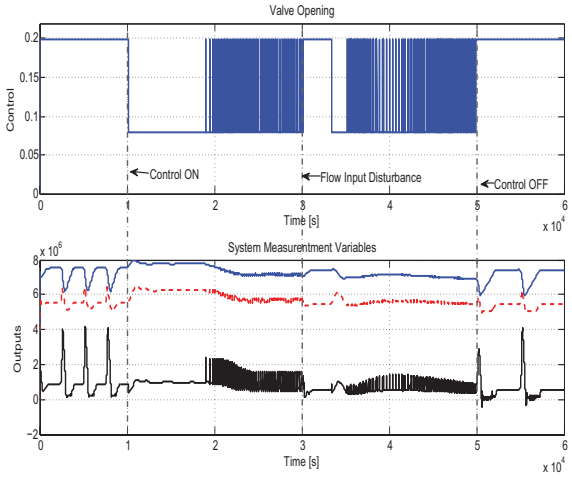


Fig. 12. Control and output system responses with slug washout SMC for a disturbance in the input riser flow rate.

13. As can be seen, the propose control law stabilize the process and at the same time allow to work over the full choke range. A disadvantage to use this propose control law is the resulting

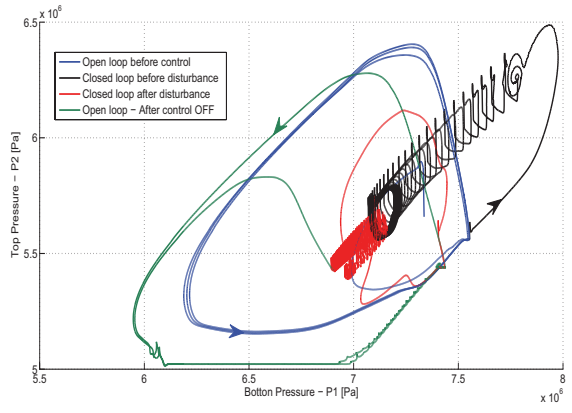


Fig. 13. State-space diagram of the system with slug washout SMC.

chattering produced by the control action signal on the choke. An alternative to overcome the high frequency chattering from the dynamics of the standard sliding mode control presented in the this Section is to design a Higher Order Sliding Mode (HOSM) control.

6. CONCLUSIONS

The lack of robustness to control slug-flow oscillations in submarine oil-risers using classical PI linear control system was tackled in this paper applying SMC techniques. Three different SMC controllers to suppress slug-flow control oscillations were proposed. Simulation results were obtained using OLGA software in order to compare the different SMC strategies subject to mass flow input riser disturbances from $5Kg.s^{-1}$ to $3Kg.s^{-1}$. The SMC technique reveal itself as a robust way of suppressing limit cycles when the mathematical model of the process is not available in practice. The dynamical of the slug system with unknown operating point was treated in this work using washout filters. This situation is manifested in the presence of mass flow rate input riser disturbances.

An existing practical obstacle to apply the standard SMC in the field is the high frequency chattering of the generated control signal. This problem leads to a premature wear down of the choke actuator and could be tackled in future works using High Order Sliding Mode - HOSM controllers.

ACKNOWLEDGEMENTS

The authors would also like to acknowledge *Scandpower* for providing an academical OLGA software license.

REFERENCES

Angulo, F., di Bernardo, M., Fossas, E., and Olivar, G. (2005a). Feedback control of limit cycle: a switching control strategy based on nonsmooth bifurcation theory. *IEEE Transactions on Circuit and Systems-I*, 52(2), 366–378.

Cunha, F.B. and Pagano, D.J. (2002). DC-DC step-up converter controlling by SMC and with an auxiliary washout filter. In *Proc. of Congresso Brasileiro de Automatica, CBA2002*. (in portuguese).

di Bernardo, M., Budd, C., Champneys, A., and Kowalczyk, P. (2008). *Piecewise-smooth Dynamical Systems: Theory and Applications*. Applied Mathematical Sciences 163. Springer.

Godhavn, J., Fard, M.P., and Fuchs, P.H. (2005). New slug control strategies, tuning rules and experiments results. *Journal of Process Control*, 15, 547–557.

Lee, H.C. and Abed, E.H. (1991). Washout filter in the bifurcation control of high alpha flight dynamics. In *Proc. of the American Control Conference*, 206–211. Boston, MA.

Storkaas, E. and Skogestad, S. (2004). Cascade control of unstable systems with application to stabilization of slug flow. *IFAC Symposium ADCHEM 2003*.

Storkaas, E. (2005). *Stabilizing control and controllability. Control solutions to avoid slug flow in pipeline-riser systems*. Ph.D. thesis, Norwegian University of Science and Technology, Norwegian.

Wang, H. and Abed, E.H. (1995). Bifurcation control of a chaotic system. *Automatica*, 31, 1213–1226.

Estimation

Oral Session

A New Process Noise Covariance Matrix Tuning Algorithm for Kalman Based State Estimators

Nina P. G. Salau*, Jorge O. Trierweiler*, Argimiro R. Secchi, Wolfgang Marquardt*****

** Federal University of Rio Grande do Sul, Chemical Engineering Department, Eng. Luiz Englert, s/n°, Campus Central, CEP 90040-040, Porto Alegre - RS, Brazil, (ninas@ enq.ufrgs.br, jorge@ enq.ufrgs.br)*

*** Federal University of Rio de Janeiro, PEQ – COPPE, Av. Horácio Macedo, 2030 - Centro de Tecnologia - Bloco G - Sala G-115, Cidade Universitária, CP: 68502, CEP 21945-970, Rio de Janeiro – RJ, (arge@ peq.coppe.ufrj.br)*

****RWTH Aachen University, Process Systems Engineering,Turmstr. 46, 52064 Aachen, Germany, (wolfgang.marquardt@avt.rwth-aachen.de)*

Abstract: A suitable design of state estimators requires a representative model for capturing the plant behavior and knowledge about the noise statistics, which are generally not known in practical applications. While the measurement noise covariance can be directly derived from the measurement device reproducibility, the choice of the process noise covariance is much less straightforward. Further, processes such as continuous process with grade transitions and batch or semi-batch process are characterized by time-varying structural uncertainties which are, in many cases, partially and indirectly reflected in the uncertainty of the model parameters. It has been shown that the robust performance of state estimators significantly enhances with a time-varying and non-diagonal process noise covariance matrix, which explicitly takes parameter uncertainty into account. For this case, the parameter uncertainty is quantified through the parameter covariance matrix. This paper presents a direct and a sensitivity method for the parameter covariance matrix computation. In the direct method, the parameter covariance matrix is found during the parameter estimation step of the SELEST algorithm, while in the sensitivity method, the parameter covariance matrix is obtained through a time-varying sensitivity matrix. The results have shown the efficacy of these methods in improving the performance of an extended Kalman filter (EKF) for a semi-batch reactor process.

Keywords: state estimator design, noise statistics, parameter estimation, sensitivity analyses.

1. INTRODUCTION

Since usually not all states of a nonlinear dynamic model are measured, they need to be estimated to be used in any control and optimization strategy. State estimators are used to estimate the unmeasured states and to filter the measured ones. Therefore, they are essential for any advanced control and optimization application. Besides an accurate plant model, an appropriate choice of process and measurement noise covariances is crucial in applying state estimators. The measurement error covariance matrix is usually known from the error statistics of the measurement device and is readily available. However, in actual problems, the process-noise statistics are often unknown, do not satisfy the assumptions of normal distribution and are mostly due to the uncertainties in the model that can be either parametric or structural.

Adaptive filtering techniques estimate noise covariances from data and have been used for nonlinear systems (Mehra, 1972; Odelson et al. 2006). The methods in this field can be divided into four general categories (Mehra, 1972): Bayesian, maximum likelihood, covariance matching, and correlation techniques. Bayesian and maximum likelihood methods have fallen out of favor because of their sometimes excessive computation times. Covariance matching is the computation of the covariances from the residuals of the state estimation problem, but has been shown to give biased estimates of the true covariances. The fourth category is correlation techniques, which is the most popular for determining these covariances (Odelson et al. 2006). However, these methods assume constant noise characteristics and the availability of data required to obtain a true representation of noise statistics. For continuous or batch processes with time-varying process dynamics and operating within a wide range of process conditions, these noise statistics are time varying. The use of a fixed value of noise statistics can lead to poor filter performance and even result in filter divergence (Vallapil & Georgakis 1999, 2000, Leu & Baratti, 2000).

Valappil & Georgakis (1999, 2000) introduced two systematic approaches to be used for the calculation of a time-varying and non-diagonal process noise covariance matrix, which explicitly takes parameter uncertainty into account. The first, called linearized approach, is based on a Taylor series expansion of the nonlinear equations around the nominal parameter values, while the second, called Monte Carlo approach, accounts for the nonlinear dependence of the system on the fitted parameters by Monte Carlo simulations that can easily be performed on-line. Both approaches have been compared favorably with the traditional methods of trial-and-error tuning of EKF. For the linearized approach, the process noise covariance matrix for the filter is obtained by a procedure using the known parameter covariance matrix. The main advantage of the linearized approach is that it involves very simple algebraic calculations and can easily be executed on-line. Afterwards this approach was employed successfully in EKF-based NMPC algorithms for batch

processes (Valappil & Georgakis, 2001, 2002; Nagy and Braatz, 2003).

In this work, a new process noise covariance matrix tuning algorithm is proposed. It is an extension of the linearized approach proposed by Valappil and Georgakis (1999, 2000) with two methods for the parameter covariance matrix computation. In the direct method, the parameter covariance matrix is found during the parameter estimation step using SELEST (Secchi et al., 2006), an algorithm for automatic selection of model parameters based on an extension of the identifiability measure of Li et al. (2004). In the sensitivity method, the parameter covariance matrix is obtained through a time-varying sensitivity matrix. Both methods can be successfully applied for state estimator design.

2. PROBLEM FORMULATION AND SOLUTION STRATEGIES

2.1 Hybrid Extended Kalman Filter (H-EKF)

Consider the following nonlinear dynamic system to be used in the state estimator

$$\begin{aligned} \dot{x} &= f(x,u,t,p)+\omega(t) \\ y_k &= h_k(x_k,t_k)+\nu_k \\ \omega(t) &\sim (0,Q) \\ \nu_k &\sim (0,R_k) \end{aligned} \tag{1}$$

where u denotes the deterministic inputs, x denotes the states, and y denotes the measurements. The process-noise vector, ω(t), and the measurement-noise vector, ν_k, are assumed to be a white Gaussian random process with zero mean and covariance Q and R_k, respectively. The H-EKF formulation uses a continuous and nonlinear model for state estimation, linearized models of the nonlinear system for state covariance estimation, and discrete measurements (Simon, 2006). This is often referred to as continuous-discrete extended Kalman filter (Jazwinski, 1970). Here, the system is linearized at each time step to obtain the local state-space matrices as below:

$$F(t)=\left(\frac{\partial f}{\partial x}\right)_{x,u,t,p_{nom}}, \qquad H(t)=\left(\frac{\partial h}{\partial x}\right)_{x,u,t,p_{nom}} \tag{2}$$

The equations that compose the different steps in the H-EKF are given below.

State transition equation:

$$\hat{x}_{k|k-1}=\hat{x}_{k-1|k-1}+\int_{k-1}^{k} f(\hat{x},u,\tau,p)\,d\tau \tag{3}$$

State covariance transition equation

$$P_{k|k-1}=P_{k-1|k-1}+\int_{k-1}^{k}\left[F(\tau)P(\tau)+P(\tau)F(\tau)^T+Q\right]d\tau \tag{4}$$

Kalman gain equation:

$$K_k=P_{k|k-1}H_k^T\left[H_kP_{k|k-1}H_k^T+R_k\right]^{-1} \tag{5}$$

State update equation:

$$\hat{x}_{k|k}=\hat{x}_{k|k-1}+K_k\left[y_k-h\left(\hat{x}_{k|k-1},t_k\right)\right] \tag{6}$$

State covariance update equation:

$$P_{k|k}=\left[I_n-K_kH_k\right]P_{k|k-1}\left[I_n-K_kH_k\right]^T+K_kR_kK_k^T \tag{7}$$

2.2 Linearized Approach to Calculate the ω(t) Statistics

As introduced by Valappil & Georgakis (1999, 2000), the linearized approach to calculate the ω(t) statistics of (1) consists in assuming that the process noise vector ω(t) mostly represents the effects of parametric uncertainty. As $\dot{x}(t)$ and $\dot{x}_{nom}(t)$ are desired to be the same, ω(t) can be defined by

$$\omega(t)=f(x,u,t,p)-f(x_{nom},u,t,p_{nom}) \tag{8}$$

Performing a first-order Taylor's series expansion of the right-hand side of (8) around the nominal state trajectory (x_{nom}) and the nominal parameters (p_{nom}), neglecting the higher-order terms, results in following approximation

$$\begin{aligned} \omega(t)=&\left(\frac{\partial f}{\partial x}\right)_{x_{nom},u,t,p_{nom}}\left[x(t)-x_{nom}(t)\right] \\ &+\left(\frac{\partial f}{\partial p}\right)_{x_{nom},u,t,p_{nom}}\left[p-p_{nom}\right] \end{aligned} \tag{9}$$

Assuming that $\left[x(t)-x_{nom}(t)\right]=\left[\hat{x}(t)-x_{nom}(t)\right]\approx 0$, the process noise is calculated from

$$\omega(t)=F_{p_{nom}}(t)\left[p-p_{nom}\right] \tag{10}$$

where $F_{p_{nom}}(t)=\left(\frac{\partial f}{\partial p}\right)_{\hat{x},u,t,p_{nom}}$. Calculating the expected value of both sides of (10) yields

$$\bar{\omega}(t)=F_{p_{nom}}(t)(p_{nom}-p_{nom})=0 \tag{11}$$

indicating that the noise sequence ω(t) has zero mean if the employed linearization in the parameters was accurate. Then the desired computation of the covariance Q(t) of ω(t) is given by

$$Q(t)=F_{p_{nom}}(t)C_pF_{p_{nom}}^T(t) \tag{12}$$

where $C_p\in\Re^{n_p\times n_p}$ is the parameter covariance matrix. In this work, we propose two methods to calculate C_p, which are presented in the next subsection.

2.3 Proposed Methods to Calculate the C_p Matrix

Consider the general process model.

$$\dot{x}=f(x,u,t,p_{nom}) \tag{13}$$

Differentiating (13) with respect to the nominal parameter vector, p_{nom}, gives

$$\dot{S}=\left(\frac{\partial f}{\partial x}\right)S+\left(\frac{\partial f}{\partial p_{nom}}\right)=F(t)S+F_{p_{nom}}(t) \tag{14}$$

where S is the sensitivity matrix $(\partial x/\partial p_{nom})$ determined by numerical integration of (14) along with the model of (13).

2.3.1 Direct Method: Parameter Covariance Matrix via Parameter Estimation

In this method, C_p is constant and directly obtained from the parameter estimation procedure. For this purpose, we have selected the SELEST algorithm proposed by Secchi et al. (2006). This algorithm uses a sensitivity matrix, S, based on calculation of the parameters effects on the measured outputs and of a linear-independence metric, as proposed by Li et al. (2004). A predictability degradation index and a parameter correlation degradation index are used as stopping criterion. The definitions of these indexes as well as the SELEST algorithm are presented in Secchi et al. (2006).

2.3.2 Sensitivity Method: Parameter Covariance Matrix via Sensitivity Matrix

As pointed out by Sharma & Arora (1993), the sensitivity matrix, S, can play a role in quantifying how good the estimate of the parameters is. For uncorrelated and normally distributed measurement errors and for nonlinear least-squares problems, a parameter covariance matrix C_p can be estimated from

$$C_p \approx s^2 \left(S^T S\right)^{-1} \tag{15}$$

where s^2 accounts for the accuracy of the data used to fit the parameters ($\hat{Y}$) and is usually represented by the residual mean square

$$s^2 = \frac{\sum\left(\hat{Y}\text{-}Y_p\right)^T\left(\hat{Y}\text{-}Y_p\right)}{n\text{-}np} \tag{16}$$

where Y_p is the estimated data, n is the number of samples and np is the number of estimated parameters. The residual mean square s^2 is also obtained from the parameter estimation using SELEST algorithm. Since the sensitivity matrix, S, is time-varying, C_p is also time-varying, which represents an advantage of this method.

3. MATRIX Q TUNING ALGORITHM

This section presents an algorithm for tuning of the process noise covariance matrix. As mentioned earlier, this algorithm is an extension of the linearized approach proposed by Valappil and Georgakis (1999, 2000) with two methods for the parameter covariance matrix computation.

Since any model is an abstraction of reality, both the structural and parametric uncertainties are present to some degree in most real situations. The structural uncertainties is often captured by uncertainty in the model parameters only. The proposed algorithm requires knowledge about which parameters can be considered time-varying. Afterwards, SELEST algorithm estimates the best possible subset of parameters within a full set of model parameters assumed time-varying.

Parameter estimation is a key ingredient to quantify the parametric model uncertainty. However, most contributions on parameter estimation in process control assume that all model states are measured, which is not true in practical applications. In order to carry out proper parameter estimation, the EKF is used in a previous stage with the nominal parameters to estimate the unmeasured states and to filter the measured ones. Afterwards, the state estimation is carried out a posteriori with the estimated parameters, p_{est}, and a time-varying and non-diagonal matrix Q obtained from the procedure described above. The required covariance matrix C_p is calculated by one of the two methods proposed in this work. The structure of the algorithm for process noise covariance matrix Q tuning is shown in Fig. 1.

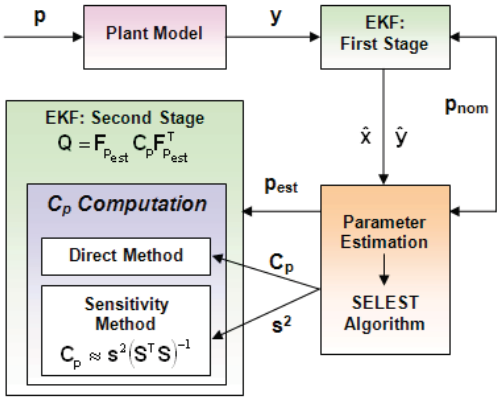


Fig. 1: Process noise covariance matrix Q tuning algorithm.

Note that the matrix Q in our algorithm takes into account the estimated parameters, p_{est}, rather than the nominal parameters, p_{nom}, as introduced by Vallapil and Georgakis (1999, 2000) in (12).

4. CASE STUDY: WILLIAMS-OTTO SEMI-BATCH REACTOR

A description of the Williams-Otto semi-batch reactor, as introduced by Forbes (1994), is provided in this section. The following reactions take place in the reactor:

$$A + B \xrightarrow{k_1} C$$
$$C + B \xrightarrow{k_2} P + E$$
$$P + C \xrightarrow{k_3} G$$

Reactant A is already present in the reactor, whereas reactant B is fed continuously to the reactor. During the exothermic reactions the products P and E as well the side-product G are formed. The heat generated through the exothermic reaction is removed by a cooling jacket, which is controlled by manipulating the cooling water temperature. The manipulated control variables of this process are the inlet flow rate of reactant B (F_B) and the cooling water temperature (T_w), whose values have been kept constant in our study. The model equations are given below and the model parameters are reported in Table 1:

$$\frac{dm_A}{dt} = -\frac{M_A}{M_A} r_1 V \tag{17}$$

$$\frac{dm_B}{dt} = F_B - \frac{M_B}{M_A} r_1 V + \frac{M_B}{M_B} r_2 V \tag{18}$$

$$\frac{dm_C}{dt} = \frac{M_C}{M_A} r_1 V - \frac{M_C}{M_B} r_2 V - \frac{M_C}{M_C} r_3 V \tag{19}$$

$$\frac{dm_P}{dt} = \frac{M_P}{M_B} r_2 V - \frac{M_P}{M_C} r_3 V \quad (20)$$

$$\frac{dm_E}{dt} = \frac{M_E}{M_B} r_2 V \quad (21)$$

$$\frac{dm_G}{dt} = \frac{M_G}{M_C} r_3 V \quad (22)$$

$$\frac{dV}{dt} = \frac{F_B}{\rho} \quad (23)$$

$$\frac{dT_r}{dt} = \frac{H - F_B c_p T_r}{V \rho c_p} \quad (24)$$

where

$$V = \frac{m_A + m_B + m_C + m_P + m_E + m_G}{\rho} \quad (25)$$

$$r_1 = k_1 \frac{m_A m_B}{V^2}; \quad r_2 = k_2 \frac{m_B m_C}{V^2}; \quad r_3 = k_3 \frac{m_C m_P}{V^2} \quad (26)$$

$$k_i = A_i e^{\frac{-E_i}{(T_r + T_{ref})}}; \quad i = 1,2,3 \quad (27)$$

$$H = F_B c_p T_{in} - \Delta H_1 \frac{M_A}{M_A} r_1 V - \Delta H_2 \frac{M_B}{M_B} r_2 V - \Delta H_3 \frac{M_C}{M_C} r_3 V - V \frac{A_0}{V_0} U(T_r - T_w) \quad (28)$$

Table 1. Model Parameters

M_A	100 kg.kmol^{-1}	ΔH_1	-263.8 kJ.kg^{-1}
M_B	200 kg.kmol^{-1}	ΔH_2	-158.3 kJ.kg^{-1}
M_C	200 kg.kmol^{-1}	ΔH_3	-226.3 kJ.kg^{-1}
M_P	100 kg.kmol^{-1}	A_0	9.2903 m^2
M_E	200 kg.kmol^{-1}	V_0	2.1052 m^3
M_G	300 kg.kmol^{-1}	U	0.23082 kJ(m^2.°C.s)$^{-1}$
A_1	1.6599E3 m^{3}kg^{-1}s^{-1}	F_B	5.7840 kg.s^{-1}
A_2	7.2117E5 m^{3}kg^{-1}s^{-1}	T_w	100 °C
A_3	2.6745E9 m^{3}kg^{-1}s^{-1}	$m_A(t_0)$	2000 kg
E_1	6666.7 K	$m_B(t_0)$	0
E_2	8333.3 K	$m_C(t_0)$	0
E_3	11111.1 K	$m_P(t_0)$	0
T_{ref}	273.15 K	$m_E(t_0)$	0
T_{in}	35 °C	$m_G(t_0)$	0
c_p	4.184 kJ.kg^{-1}.°C^{-1}	$V(t_0)$	2 m^3
ρ	1000 kg.m^{-3}	$T_r(t_0)$	65 °C
t_f	1000 s		

In order to illustrate the application of the Q tuning algorithm, the kinetic parameters E_1, E_2, and E_3 were chosen as uncertain parameters. A parametric uncertainty of ±5% is assumed. The correct parameter values ("plant parameters"), p, and the nominal parameters, p_{nom}, are reported in Table 2.

Table 2. Uncertain Parameters

	E_1	E_2	E_3
p	6333.4	7916.3	11666.6
p_{nom}	6666.7	8333.3	11111.1

The application of Q tuning algorithm to the Williams-Otto semi-batch reactor is shown below.

4.1 First Iteration of Q Tuning Algorithm.

4.1.1 Results of State Estimation: First Stage

A first state estimation with nominal parameters is performed to provide information on unmeasured states to be used in the subsequent parameter estimation step. The states and measurements of the Williams-Otto semi-batch reactor are

$$x = [m_A \quad m_B \quad m_C \quad m_P \quad m_E \quad m_G \quad V \quad T_r] \quad (29)$$

$$y = [m_B \quad m_E \quad m_G \quad V] \quad (30)$$

The measurements are obtained from a simulation of the plant model with the plant parameters p. The initial condition and the parameters of the state estimation are

$$x_0 = [2000 \quad 0 \quad 0 \quad 0 \quad 0 \quad 0 \quad 2 \quad 65] \quad (31)$$

$$P_0 = 0.0001^2 I_{8x8} \quad (32)$$

$$\Delta t = t_k - t_{k-1} = 31.25 \quad (33)$$

$$R = diag(0.1^2 \quad 0.1^2 \quad 0.01^2 \quad 0.01^2) \quad (34)$$

$$Q = diag(0.1^2 \quad 0.01^2 \quad 0.1^2 \quad 0.01^2 \quad 0.01^2 \quad 0.1^2 \quad 0.1^2 \quad 0.01^2) \quad (35)$$

The state estimation results using the EKF with nominal parameters and a constant-value and diagonal matrix Q are shown in Fig. 2. As expected, in the presence of a constant parametric model mismatch, the estimated states show a bias (cf. Fig. 2b).

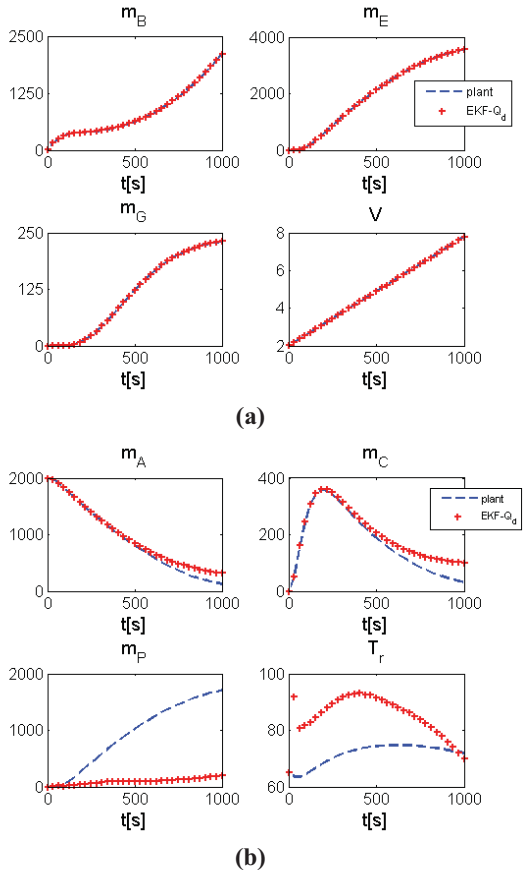


Fig. 2: EKF with nominal parameters (p_{nom}) and a constant-value and diagonal matrix Q (Q_d): (a) filtered measured states and (b) estimated states.

4.1.2 Results of Parameter Estimation Step

Using the SELEST algorithm, the parameter estimation step is based on the nominal parameters, p_{nom}. The data used to fit the parameters are composed of the estimated states and the filtered measured states provided by the first state estimation stage. As a result of the parameter estimation step, the SELEST algorithm provides the estimated parameters, p_{est}, the parameter covariance matrix, C_p, and the residual mean square, s^2

$$p_{est} = \begin{bmatrix} 6333.8 & 7957.9 & 11223.5 \end{bmatrix}$$

$$C_p = \begin{bmatrix} 3.2469E-5 & -8.6175E-6 & -4.2417E-5 \\ -8.6175E-6 & 3.8060E-5 & 8.2221E-6 \\ -4.2417E-5 & 8.2221E-6 & 3.8060E-5 \end{bmatrix}$$

$$s^2 = \frac{\sum\left(\hat{Y}-Y_p\right)^T\left(\hat{Y}-Y_p\right)}{\left(\frac{t_f}{\Delta t}+1\right)-np} = \frac{\sum\left(\hat{Y}-Y_p\right)^T\left(\hat{Y}-Y_p\right)}{\left(\frac{1000}{31.25}+1\right)-3} = 1.66E2$$

where $\hat{Y}$ is composed of the estimated and the filtered measured states resulting from the first state estimation stage and Y_p is calculated by the SELEST algorithm.

4.1.3 Results of State Estimation: Second Stage

At this point, state estimation is carried out with the estimated parameters and the time-varying and non-diagonal Q obtained by C_p. The performance of the EKF with the following choices for calculating Q is compared and the results are shown in Fig. 3.

Direct method: Q is time-varying and non-diagonal, with p_{est} and C_p estimated by means of the SELEST algorithm.

Sensitivity method: Q is time-varying and non-diagonal, with p_{est} and s^2 estimated by means the SELEST algorithm and C_p obtained via sensitivity integration.

Random Variation: Proposed by Valappil and Georgakis (1999, 2000). The parameters in the plant are assumed to vary with time, taking values at each sample interval from a nominal distribution. The mean value of the varying plant parameter is assumed to be different from the nominal value of the model parameter by a fixed amount σ. The parameter covariance matrix used in the filter is given by $C_p = \sigma^2$.

Monte Carlo Approach: Proposed by Valappil and Georgakis (1999, 2000). This approach accounts for the nonlinear dependence of the system on the fitted parameters by Monte Carlo simulations. For the case study, 500 Monte Carlo simulations of different parameter values were used, resulting in 500 evaluations of the process noise.

The initial conditions (31) and the parameters of the state estimation algorithm (32 to 35) remain the same in this stage. According to Fig. 3, the EKF with a time-varying and non-diagonal matrix Q obtained by random variation in the plant parameters presents the worst performance. The sensitivity method performs better compared to the direct method. As mentioned before, an advantage of this method is that the parameter covariance matrix, C_p, is time-varying due to the time-varying sensitivity matrix S. In spite of accounting the nonlinear dependence of the system on the fitted parameters, the Monte Carlo shows a performance slightly inferior to the that of the sensitivity method for estimated states (Fig. 3b) and a performance quite inferior to that of the sensitivity and direct methods for measured states (Fig. 3a), not to mention the high computational effort.

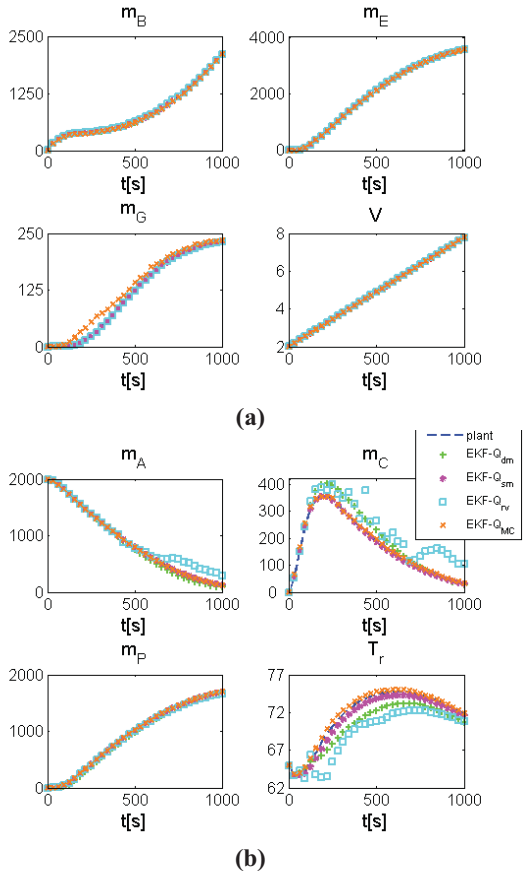


Fig. 3: EKF with estimated parameters (p_{est}) and a time-varying and non-diagonal Q matrix obtained by the proposed methods: direct (Q_{dm}) and the sensitivity (Q_{sm}); and by the literature methods: random variation (Q_{rv}) and Monte Carlo (Q_{MC}): (a) filtered measured states and (b) estimated states.

4.2 Second Iteration of Matrix Q Tuning Algorithm

Since the unmeasured states are unknown in practical applications, the state estimation accuracy shall be quantified. The matrix Q tuning algorithm is hence performed iteratively with the proposed methods for the parameter covariance matrix computation until the state estimation accuracy could not be significantly improved.

The parameter estimation is now taking place with the estimated parameters, p_{est}, and the estimated states and filtered measurements from the first iteration of the proposed algorithm. The parameter estimation results for both methods are given in Table 3.

Table 3. Uncertain Parameters

Method	p_{est}			s^2
	E_1	E_2	E_3	
Direct	6328.1	7952.7	11668.9	7.0880
Sensitivity	6333.9	7921.3	11666.3	0.7785

Disregarding numerical round off, the matrix C_p is the same for both methods, i.e.

$$C_p = \begin{bmatrix} 4.9337E-5 & -4.6902E-6 & 4.3956E-6 \\ -4.6902E-5 & 7.7390E-5 & 4.7829E-6 \\ 4.3956E-5 & 4.7829E-6 & 5.0158E-5 \end{bmatrix}$$

As expected, the residual mean square s^2 is smaller for the sensitivity method which performs better than direct method in an a-posteriori state estimation stage, as shown in Fig. 4.

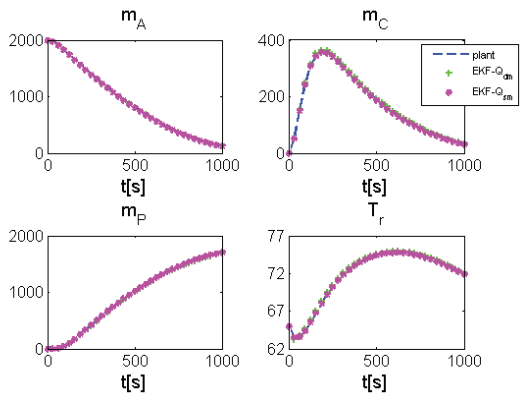


Fig. 4: Estimated states for the EKF with estimated parameters (p_{est}) and a time-varying and non-diagonal Q matrix obtained by the direct (Q_{dm}) and the sensitivity (Q_{sm}) methods.

For this example, a third iteration of the algorithm has not improved significantly the state estimation accuracy.

5. CONCLUSIONS

A new process noise covariance matrix tuning algorithm is presented which incorporates the linearized approach proposed by Valappil and Georgakis (1999, 2000) with two methods for the parameter covariance matrix computation. As pointed out by Valappil and Georgakis (1999, 2000), the investment in a nondiagonal time-varying matrix Q is justified because (a) parametric uncertainties cause significant cross-correlations between the process noises for different states (b) for continuous or batch processes with time-varying process dynamics and operating on wide range of process conditions, the noise statistics are time varying.

The Q tuning algorithm consists of two state estimation steps and a parameter estimation step in between. A first state estimation step with nominal parameters is performed to provide information on unmeasured states to be used in the subsequent parameter estimation step. Afterwards, the state estimation is carried out with the estimated parameters and a time-varying and non-diagonal tuning of matrix Q obtained from the parameter covariance matrix C_p, evaluated by the direct and sensitivity methods. In the direct method, C_p is assumed to be constant and directly obtained from the parameter estimation step using the SELEST algorithm (Secchi et al., 2006). In the sensitivity method, C_p is obtained from the computation of the time-varying sensitivity matrix. Although the EKF with a time-varying and non-diagonal matrix Q obtained from the sensitivity method performs better compared to the direct method, both methods can be successfully applied for state estimator design. Moreover, these methods improve considerably the EKF performance when compared to a) the case of a constant-value and diagonal matrix Q in the presence of constant parametric uncertainty and to b) the methods of prior publications. Successive iterations of the Q tuning algorithm shall improve the state estimation accuracy. For the Williams-Otto semi-batch reactor, only two iterations were necessary to improve the state estimation accuracy, significantly.

The main advantage of the algorithm presented in this work is that it is feasible for practical applications. Besides, of an online EKF tuning, the process model is updated online due to the integration of the state and the parameter estimation steps. Further, the algorithm eliminates an offline, exhaustive, and inexact tuning of EKF by trial and error.

REFERENCES

Forbes, J.F. (1994). Model Structure and Adjustable Parameter Selection for Operations Optimizations. *PhD thesis*, McMaster University, Hamilton, Canada, 1994.

Jazwinski, A. H. (1970). *Stochastic Processes and Filtering Theory*, Academic Press, New York.

Leu, G..; Baratti, R. (2000). An Extended Kalman Filtering Approach with a Criterion to set its Tuning Parameters; Application to a Catalytic Reactor. *Computers & Chemical Engineering*, 23, 1839-1849.

Li, R.; Henson, M.A.; Kurtz, M.J. (2004). Selection of model parameters for off-line parameter estimation. *IEEE Transactions on Control Systems Technology*, 12 (3), 402-412.

Nagy, Z. K.; Braatz, R.D. (2003). Robust nonlinear model predictive control of batch processes. *AIChE J.*, 49(7), 1776- 1786.

Odelson, B.J.; Lutz, Alexander, L; Rawling, J.B. (2006). The autocovariance least-squares method for estimating covariances: Application to model-based control of chemical reactors. *IEEE Transactions on Control Systems Technology*, 14(3), 532-540.

Secchi, A.R.; Cardozo, N.S.M.; Almeida Neto, E.; Finkler, T.F. (2006). An algorithm for automatic selection and estimation of model parameters, *In: Proceedings of International Symposium on Advanced Control of Chemical Processes, ADCHEM 2006*, Gramado, Brazil.

Mehra, R.K. (1972). Approaches to adaptive filtering. *IEEE Trans. Automat. Contr.*, 17(5), 693-698.

Sharma, M.S.; Arora, N.D. (1993). Optima: A nonlinear model parameter extraction program with statistical confidence region algorithms, *IEEE Trans. Comp. Aided Des. Int. Circuits Syst*., 12, 982–987.

Simon, D. (2006). *Optimal State Estimation: Kalman, H Infinity, and Nonlinear Approaches*, Wiley-Interscience, New Jersey.

Valappil, J.; Georgakis, C. (1999). A systematic tuning approach for the use of extended Kalman filters in batch processes. *Proc. of the American Control Conf., IEEE Press,* Piscataway, NJ, 1143.

Valappil, J.; Georgakis, C. (2000). Systematic estimation of state noise statistics for extended Kalman filters. *AIChE J.*, 46(2), 292-398.

Valappil, J.; Georgakis, C. (2001). A systematic tuning approach for the use of extended Kalman filters in batch processes. *Proc. of the American Control Conf., IEEE Press,* Piscataway, NJ, 99.

Valappil, J.; Georgakis, C. (2002). Nonlinear model predictive control of end-use properties in batch reactors. *AIChE J.,* 48(9), 2006-2021.

Observer Design for Systems with Continuous and Discrete Measurements ⋆

C.P. Guillén-Flores * **B. Castillo-Toledo** *
J.P. García-Sandoval ** and **V. González-Álvarez** **

* *CINVESTAV-IPN, Av. Cientifica 1145 , colonia el Bajío, Zapopan , 45015, Jalisco, México (e-mail: [cguillen, toledo]@gdl.cinvestav.mx)*
** *Chemical Engineering Department, University of Guadalajara, Calz. Gral. Marcelino García Barragán 1451, Guadalajara, Jalisco 44430, México (e-mail: paulo.garcia@cucei.udg.mx)*

Abstract: Classical observers are constructed on the basis of the nature of the measurement signals, namely, a continuous observer requires continuous output measurements. In this work, a novel observer which estimates continuous states when continuous and discrete measurements are available is presented. By resetting the initial condition of the observer at each sample instant, the convergence of the continuous states is guaranteed. The application to the the estimation of substrate and biomass concentrations in an anaerobic wastewater treatment process in which continuous and discrete measurements usually appear, shows the feasibility of the proposed scheme.

Keywords: jump observer, anaerobic digestion, discrete measurements

1. INTRODUCTION

Because of the increasing complexity and necessity for safety of industrial processes, efficient monitoring, decision and control systems are becoming more an more important. This is particularly true in the case of bioprocesses where the state of the live organisms of the system must be closely monitored. Extensive surveys have been published on this topic (Dochain, 2008). Furthermore, the last two decades have seen an increasing interest in improving the operation of bioprocesses by applying advanced control schemes. In particular, biological waste treatment process, more efficient that the traditional physicochemical methods but at the same time more complex, call for a consistent good performance,which leads to a needing for more efficient instrumentation, control and automation.

To apply any control strategy it is necessary to measure the process main variables, this can be performed placing sensors (**?**), however, although in many cases continuous measurements are easily available, for example the temperature or pH, due to economical reasons or consuming time techniques, other key variables can be only measured intermittently, or even not measured at all. For this reason the non measurable state variables should be estimated from available measurements (Meleiro and Filho, 2000). To deal with these problems, many solutions have been proposed in the past such as the well known classical Kalman filters and Luenberger observers (Ray, 1980) in both, continuous and discrete approaches. On of the reasons for the popularity of these estimators is that they are easy to implement since the algorithm can be derived directly from the state space model. However, these state observers can not be easily implemented when both continuous and discrete information must be considered. In this direction Scali et al. (1997) have proposed an extended Kalman filter which update some observer parameters each time that the sampled date is available. Using Lyapunov functions , Liu et al. (2008) and Muñoz de la Peña and Christofides (2008) have designed controllers that involve continuous and discrete retarded measurements. Nguang and Shi (2003) also use discrete measurements to design continuous fuzzy control algorithms. Based on this idea in this work it is proposed a continuous observer to be continuously updated from the continuous measurements and also retune the states at each instant when the discrete measurement are available.

This work is organized as follows. A review of jump observers is presented in section 2, then in section 3 the observational problem is formulated, while the proposed solution is developed in section 4. In section 5 we analyze the dynamic behavior through numerical simulations for an anaerobic digestion system. Finally we close the paper with some concluding remarks.

2. BASIC FACTS OF JUMP OBSERVERS

Consider the linear system

$$\dot{x}(t) = Ax(t) + Bu(t) \qquad \forall t \in [0, \infty) \tag{1}$$

$$y(k\delta) = Cx(k\delta) \qquad k = 1, 2, 3, \ldots, \tag{2}$$

where $x \in \mathbb{R}^n$, $u \in \mathbb{R}^m$, and $y \in \mathbb{R}^q$ are the state, input and output vectors, respectively. In this case the outputs are obtained at each sampling time δ.

The usual way to estimate the unknown states of system (1) from output (2) consists in discretizing the system

⋆ Partially Supported by PROMEP under proyect 103.5/08/2919 and CONACYT under grant 43658.

and design a discrete observer. However, the observer thus obtained provides only information at each sampling period. Additionally, to obtain a discrete version of (1) it is necessary to have a well defined input in order to place the appropriate holder (for example a zero holder or a exponential holder), hence unexpected input variations during intersampling periods may produce discrete observer failures (García-Sandoval, 2006). For this reason, an interesting problem would be to construct an observer given by

$$\dot{z}(t) = Az(t) + Bu(t) \qquad \forall t \neq k\delta \tag{3}$$

$$z\left(k\delta^{+}\right) = z(k\delta) - G\left[y(k\delta) - Cz(k\delta)\right] \quad t = k\delta \tag{4}$$

where $z \in \mathbb{R}^n$ are the observer states and $z\left(k\delta^{+}\right)$ denotes the updated observer states at each sampling instant. This is a continuous observer which updates its states at each sampling instant. The next lemma establishes conditions for the existence of such observer.

Lemma 1. Consider system (1)-(2) and suppose the pair $\left(e^{A\delta}, C\right)$ is observable, then an observer of the form (3)-(4) with the matrix gain G such that matrix $(I + GC)\, e^{A\delta}$ is Schur, guaranteeing that $\lim_{t\to\infty}\left[x(t) - z(t)\right] = 0$.

Proof. See Appendix.

Remark 2. The main feature of observer (3)-(4) remains in the fact that the intersampling state information is available at any time and it is not necessary to have a pre-established dynamic behavior for the input. Equation (3) can be seen as a continuous open loop observer in the intersampling period and whose states, according to (4), are reseted each sampling period.

3. PROBLEM FORMULATION

Consider the dynamic system

$$\dot{x}(t) = f\left(x(t), u(t)\right) \quad \forall t \in [0, \infty) \tag{5a}$$

$$y_1(t) = C_1 x(t) \qquad \forall t \in [0, \infty) \tag{5b}$$

$$y_2(k\delta) = C_2 x(k\delta) \qquad k = 1, 2, 3, \ldots \tag{5c}$$

where $x \in \mathbb{R}^n$, $u \in \mathbb{R}^m$ and $y_1 \in \mathbb{R}^{q_1}$, $y_2 \in \mathbb{R}^{q_2}$ are the state, input and output vectors for the dynamic system, respectively. The outputs are divided into continuous (y_1), and discrete (y_2) with sampling time δ. For this system it is desirable to design a continuous observer which uses both discrete and continuous measurements, in order to have continuous information about the full vector state. The following assumption is instrumental for the observer design.

Assumption 3. Defining

$$A = \left.\frac{\partial f}{\partial x}\right|_{x=0,u=0} \quad \text{and} \quad B = \left.\frac{\partial f}{\partial u}\right|_{x=0,u=0}$$

as the linear matrices for system (5), it is assumed that the pair (A, C), with

$$C = \begin{pmatrix} C_1 \\ C_2 \end{pmatrix}$$

is observable but, the pairs (A, C_1) and (A, C_2) related with continuous and discrete measurements, are not necessarily completely observable. That is, the observability matrix of these pairs may not have full rank.

In the following section it is presented a continuous observer for system (5), which is the main result of this work.

4. OBSERVER DESIGN

Assume that there is a transformation $T \in \mathbb{R}^{n\times n}$, such that the linear approximation of system (5a)-(5b) becomes

$$\begin{aligned} \dot{z} &= \bar{A}z(t) + \bar{B}u(t) \\ y_1 &= \bar{C}_1 z(t) \end{aligned} \tag{6}$$

where

$$z = Tx = \begin{pmatrix} z_1 \\ z_2 \end{pmatrix}, \quad \bar{A} = TAT^{-1} = \begin{pmatrix} A_{11} & 0 \\ A_{21} & A_{22} \end{pmatrix}$$

$$\bar{B} = TB = \begin{pmatrix} B_1 \\ B_2 \end{pmatrix}, \quad \bar{C}_1 = C_1 T^{-1} = (C_{11}\ 0)$$

$z_1 \in \mathbb{R}^{n_1}$, $z_2 \in \mathbb{R}^{n_2}$, and the pair (A_{11}, C_{11}) is completely observable. In this case a partial observer for z_1 can be designed in such way that given a matrix G_{11}, $(A_{11} - G_{11}C_{11})$ is Hurwitz. Applying the inverse transformation, the proposed observer is,

$$\dot{\zeta}(t) = f\left(\zeta(t), u(t)\right) - G_1\left(C_1\zeta(t) - y_1(t)\right)$$

where

$$G_1 = T^{-1}\begin{pmatrix} G_{11} \\ 0 \end{pmatrix}.$$

This is a partial observer which only make use of continuous measurements (5b), however, using discrete measurements it is possible to design a jump observer as described in section 2, which include both continuous and discrete measurement. The following theorem states this result.

Theorem 4. Consider the system (5), which has a set of continuous measurements (5b) and a set of discrete measurements (5c) with sampling time δ. Furthermore consider that there is a transformation $T \in \mathbb{R}^{n\times n}$ which transforms the linear approximation of system (5a)-(5b) to its observable canonical form (6), while the matrix $G_1 = T^{-1}\left(G_{11}^T\ 0\right)^T$, is calculated in such a way that $(A_{11} - G_{11}C_{11})$ is Hurwitz and the matrix G_2 is such that $(I + G_2 C)\, A_d$ is Schur, with $A_d = e^{(A - G_1 C_1)\delta}$. Then, an observer for system (5), which takes continuous measurements and is also updated each sampling period is given by

$$\begin{aligned} \dot{\zeta}(t) &= f\left(\zeta(t), u(t)\right) && \forall t \neq k\delta \\ &\quad -G_1\left(C_1\zeta(t) - y_1(t)\right), \end{aligned} \tag{7a}$$

$$\begin{aligned} \zeta\left(k\delta^{+}\right) &= \zeta(k\delta) && t = k\delta, \\ &\quad +G_2\left(C\zeta(k\delta) - y(k\delta)\right), && k = 1, 2, 3, \ldots \end{aligned} \tag{7b}$$

where $\zeta \in \mathbb{R}^n$ are the observer states and $\zeta\left(k\delta^{+}\right)$ are its updated values at each sampling time and

$$y(k\delta) = \begin{pmatrix} y_1(k\delta) \\ y_2(k\delta) \end{pmatrix}.$$

This observer guarantees that, in a neighborhood of the origin, the error between the system and the observer states tends asymptotically to zero, i.e. $\lim_{t\to\infty}[x(t) - \zeta(t)] = 0$.

Proof. First, consider the linear approximations of both, system (5a) and observer (7a)

$$\dot{x}(t) = Ax(t) + Bu(t) \quad \forall t \in [0, \infty) \tag{8}$$

$$\dot{\zeta}(t) = (A - G_1C_1)\zeta(t) + Bu(t) + G_1y_1(t) \tag{9}$$

additionally, consider that there is a matrix T that transforms the system (8) and its output (5b) to its observable canonical form (6), i.e.

$$z = Tx = \mathrm{col}(z_1, z_2),$$
$$\xi = T\zeta = \mathrm{col}(\xi_1, \xi_2),$$

where z_1 and ξ_1 are the observable modes of x and ζ. Then for the observable subsystems of z and ξ defining the error $e_1(t) = z_1(t) - \xi_1(t)$, whose dynamic is

$$e_1(t) = (A_{11} - G_{11}C_{11})e_1(t).$$

Since G_{11} is such that $(A_{11} - G_{11}C_{11})$ is Hurwitz, $e_1(t)$ tends asymptotically to zero. On the other hand, using discrete measurements, $y(k\delta)$, a jump observer (7) which allows the updating of the continuous dynamic observer states in every sampling period it is designed, taking advantage of the discrete information. Defining now the error

$$\eta(t) = x(t) - \zeta(t)$$
$$\eta(k\delta^+) = x(k\delta) - \eta(k\delta^+)$$

its linear dynamic approximation around $\eta = 0$ is

$$\dot{\eta}(t) = (A - G_1C_1)\eta(t) \qquad \forall t \neq k\delta$$
$$\eta(k\delta^+) = (I + G_2C)\eta(k\delta) \qquad t = k\delta, \quad k = 1, 2, 3, \ldots$$

As described in Lemma 1, these dynamics are stable if the pair (A_d, C) with $A_d = e^{(A-G_1C_1)\delta}$ is observable and the gain G_2 is such that the matrix $(I + G_2C)A_d$ is Schur, thereby ensuring that $\lim_{k\to\infty}[x(k\delta) - \zeta(k\delta)] = 0$ and thus $\lim_{t\to\infty}[x(t) - \zeta(t)] = 0$, which proves the theorem.

Observer (7) can be seen as a hybrid observer since incorporates continuous dynamics (7a) and a discrete event (7b) which modifies the continuous part. It should be also noted that the calculation of the observer part for continuous measurements is independent of the discrete observer part, however, the total discrete observer depends on the gain G_1.

5. STUDY CASE

Last years, the environmental laws have been tightened and it has became mandatory treating wastewater from industries as well households (**?**Huntington, 1998). Because of this, the wastewater treatment control processes have received great importance, especially anaerobic processes are being widely considered as an alternative for the treatment of wastewater because it produces smaller quantities of organic matter and also yields a high-energy gas (Méndez-Acosta et al., 2008). To achieve the control of these processes, state observers are frequently used, however for economical reasons some key variables can just be measured using long sampling times, while others may be measured more often. For this reason, a jump observer is proposed as presented in theorem 4.

There exists many dynamic models to describe anaerobic process (**?**Batstone et al., 2002; Bernard et al., 2006). However, to apply the proposed observer, a macroscopic model of the anaerobic process developed and validated by Bernard et al. (2001) it is considered,

$$\dot{X}_1 = (\mu_1(S_1) - \alpha D)X_1 \tag{10a}$$

$$\dot{S}_1 = -k_1\mu_1(S_1)X_1 + (S_{1in} - S_1)D \tag{10b}$$

$$\dot{X}_2 = (\mu_2(S_2) - \alpha D)X_2 \tag{10c}$$

$$\dot{S}_2 = -k_3\mu_2(S_2)X_2 + k_2\mu_1(S_1)X_1 + (S_{2in} - S_2)D \tag{10d}$$

where X_1, X_2, S_1, S_2, are respectively the concentrations of acidogenic bacteria, methanogenic bacteria, Chemical Oxygen Demand (COD) and Volatile Fatty Acids (VFA), D is the dilution rate, defined by the ratio $D = Q/V$, where Q is the feeding flow and V the digester volume, S_{1in} and S_{2in} are respectively the concentrations of influent organic substrate and of influent VFA. The $k_i s$ are pseudo-stoichiometric coefficients associated to the bioreactions. Parameter $\alpha \in (0, 1]$ represents the fraction of the biomass which is not retained in the digester (Hess and Bernard, 2008). The bacterial growth rates $\mu_1(S_1)$ and $\mu_2(S_2)$, are nonlinear functions given respectively by the Monod and Haldane kinetics (Henze and Harremoes, 1983)

$$\mu_1(S_1) = \mu_{\max 1}\frac{S_1}{S_1 + K_{S1}}$$

$$\mu_2(S_2) = \mu_{\max 2}\frac{S_2}{S_2 + K_{S2} + (S_2/K_{I2})^2}$$

where $\mu_{1\max}$, K_{S1}, $\mu_{2\max}$, K_{S2} and K_{I2} are the maximum bacterial growth rate and the half-saturation constant associated to the substrate S_1, the maximum bacterial growth rate in the absence of inhibition, and the saturation and inhibition constants associated to substrate S_2, respectively. The values of parameters and the input concentrations used for simulations are listed in Tables 1 and 2.

If we consider that VFA concentration (S_2) is a continuous measurement while the COD concentration (S_1) can just be periodically acquired (in fact in real operations, VFA concentration can be obtained up to every hour or less (Méndez-Acosta et al., 2008), hence it can be considered continuous compared with the resident time and the COD concentration that could be measured even just once a

Table 1. Model Parameters (Alcaraz-González et al., 2003)

Parameter	Value
μ_1	$1.2\,\mathrm{d}^{-1}$
$\mu_{\max 2}$	$0.69\,\mathrm{d}^{-1}$
K_{S1}	$4.95\,\mathrm{kg\ COD/m^3}$
K_{S2}	$9.28\,\mathrm{mol\ VFA/m^3}$
K_{I2}	$20\,\mathrm{mol\ VFA/m^3}$
k_1	6.6 kg COD/kg X_1
k_2	7.8 mol VFA/kgX_1
k_3	611.2 mol VFA/kgX_2
α	0.5 (addimentional)

Table 2. Input Concentrations

Substrate	Value
S_{1in}	20 Kg COD/m^3
S_{2in}	100 mol VFA/m^3

day), the jump observer developed in the previous section can be then applied to the dynamic system (10) writing it in the form

$$\dot{x}(t) = \begin{pmatrix} \mu_1(S_1)X_1 \\ -k_1\mu_1(S_1)X_1 \\ \mu_2(S_2)X_2 \\ -k_3\mu_2(S_2)X_2 + k_2\mu_1(S_1)X_1 \end{pmatrix} + \begin{pmatrix} -\alpha X_1 \\ S_{1in} - S_1 \\ -\alpha X_2 \\ S_{2in} - S_2 \end{pmatrix} u(t) \tag{11}$$

$$y_1(t) = (\,0\ 0\ 0\ 1\,)\, x(t)$$
$$y_2(k\delta) = (\,0\ 1\ 0\ 0\,)\, x(k\delta)$$

where

$$x(t) = \begin{pmatrix} X_1 \\ S_1 \\ X_2 \\ S_2 \end{pmatrix}, \quad C = \begin{pmatrix} C_1 \\ C_2 \end{pmatrix} = \begin{pmatrix} 0\ 0\ 0\ 1 \\ 0\ 1\ 0\ 0 \end{pmatrix},$$

and $u\,(t) = D\,(t)$. System (11) can be represented as

$$\dot{x}(t) = f(x) + g(x)u(t). \tag{12}$$

To calculate the jump observer (7) is necessary to linearize the system (12) around a neighborhood of equilibrium points (Hess and Bernard, 2008; Méndez-Acosta et al., 2008) so the system has the form

$$\dot{x}(t) = Ax(t) + Bu(t) + \hat{f}(x, u)$$

where

$$A = \left[\frac{\partial f}{\partial x} + \frac{\partial g}{\partial x}u\right]_{x=0,u=0}, \quad B = g\,(x)|_{x=0}$$

are the linear approximation matrices around the steady state [see (Hess and Bernard, 2008) for a detailed steady state analysis]. In this case, the observability matrices for pairs (A, C_1), (A, C_2) and (A, C) have ranks 4, 2 and 4, respectively, i.e. using S_2 it is possible to estimate the four states, while using S_1 it is just possible to estimate the acidogenic part of the system. This is obvious since system (10) has a cascade dynamic form between acidogenic and methanogenic dynamics.

Considering a sampling time (δ) equal to 1 and parameters listed in Tables 1 and 2, it is easy to verify that A, B and A_d take the values

$$A = \begin{pmatrix} 0 & 0.7900 & 0 & 0 \\ -1.8756 & -5.7825 & 0 & 0 \\ 0 & 0 & 0 & 0.0015 \\ 2.2166 & 6.1622 & -173.6936 & -1.4701 \end{pmatrix},$$

$$B = \begin{pmatrix} -2.7976 \\ 18.4640 \\ -0.1413 \\ 90.0000 \end{pmatrix},$$

$$A_d = \begin{pmatrix} 0.8033 & 0.1145 & 0 & 0 \\ -0.2719 & -0.0349 & 0 & 0 \\ 0.0003 & 0.0008 & 0.9186 & 0.0007 \\ 0.1556 & 0.3069 & -87.2825 & 0.1798 \end{pmatrix},$$

Using LQR techniques to calculate observer gains, observer (7) is designed in order to fulfill theorem 4, obtaining

Table 3. Initial conditions for simulations runs.

State variable	$X_1\,(0)$ kg/ m^3	$S_1\,(0)$ kg/ m^3	$X_2\,(0)$ kg/ m^3	$S_2\,(0)$ mol/ m^3
Plant	1.433	0.1	0.2	0.4
Observer	0.5	0.3	0.1	1

$$G_1 = \begin{pmatrix} -0.0005 \\ 0.0115 \\ -0.9985 \\ 17.2429 \end{pmatrix}, \quad G_2 = \begin{pmatrix} 0.0007 & 0.3923 \\ -0.0002 & -0.1330 \\ 0 & 0.0003 \\ 0 & -0.0002 \end{pmatrix}.$$

5.1 Simulation Results

In order to illustrate the performance of observer (7), some numerical simulations were carried out. Initial conditions and input concentrations for these simulations are listed in Table 3 and 2, while dilution rate was considered as a time varying sinusoidal signal around the nominal value. To verify if the incorporation of the discrete measurement to the continuous observer reduces convergence time, hybrid observer (7) was compared with a continuous observer identical to (7a) without the use of (7b) (or equivalently, for this observer G_2 was settled equal to zero). Figure 1 shows the dynamic behavior of hybrid observer, the discrete actualization is clearly visible in Figure 1a where at time $t = 1$ d there is a jump on the acidogenic biomass estimation. As can be seen, observer states converges after approximately tree days. In contrast, the continuous observer (see Figure 2) converges in approximately twelve days, i.e. four times slower than the hybrid observer. Comparing both observers it easy to see that the hybrid observer obtained a faster convergence rate.

6. CONCLUSIONS

An nonlinear observer which updates the states using continuous and discrete measurements was presented. Despite this is a local observer, since observer gain matrices were calculated using the linear approximation of the original nonlinear system, its application to an anaerobic digestion model presents an excellent performance and stability, obtaining an improvement in convergence rate in comparison with an observer which only uses the continuous information. As future work, the authors are considering to extend this theory to the case where there exists parametric variations in the original plant, as well as the use of these observers to the control of systems with continuous and discrete measurements.

(Chapter head:)*

Bibliography

Alcaraz-González, V., Harmand, J., Dochain, D., Rapaport, A., Steyer, J., Pelayo-Ortiz, C., and González-Alvarez, V. (2003). A robust asymptotic observer for chemical and biochemical reactors. In *Proc. Of the IFAC ROCOND 2003*. IFAC.

Batstone, D., Keller, J., Angelidaki, I., Kalyuzhnyi, S., Pavlostathis, S., Rozzi, A., Sanders, W., Siegrist, H., and Vavilin, V. (2002). *Anaerobic Digestion Model No. 1 (ADM1)*, volume 13 of *Scientific and Technical Report*. IWA Publishing, London.

Bernard, O., Chachuat, B., Hélias, A., and Rodríguez, J. (2006). Can we assess the model complexity for

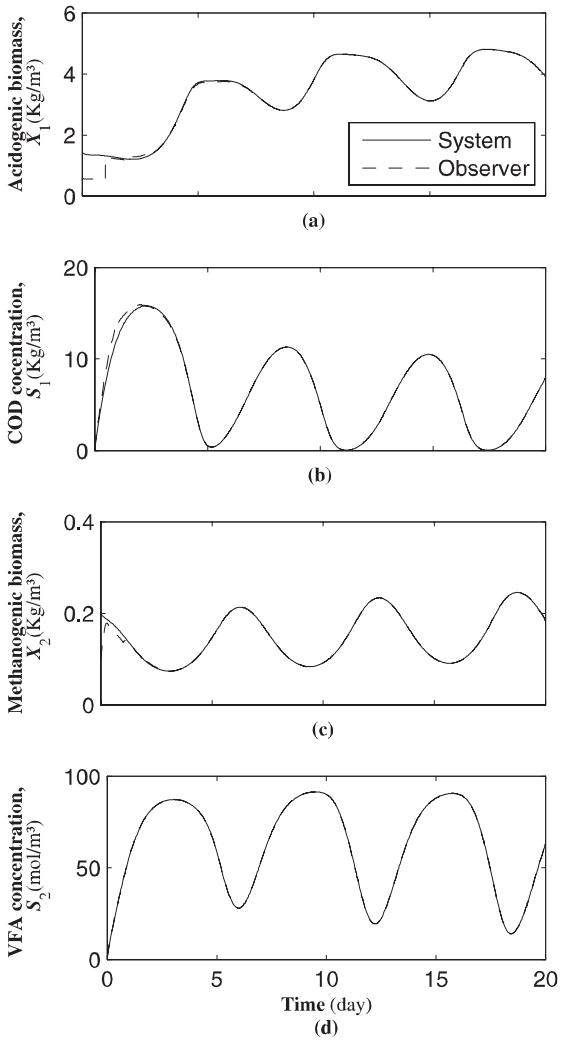


Fig. 1. Hybrid observer simulation.

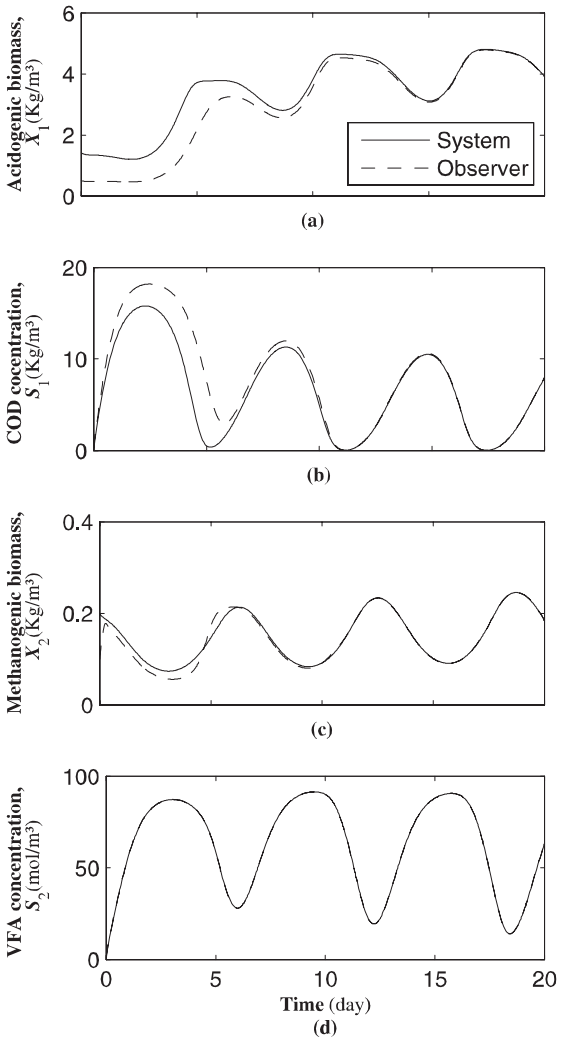


Fig. 2. Continuous observer simulation.

a bioprocess: Theory and example of the anaerobic digestion process. *Water Science Technology*, 53(1), 85–92.

Bernard, O., Hadj-Sadok, Z., Dochain, D., Genovesi, A., and Steyer, J. (2001). Dynamical model development and parameter identification for anaerobic wastewater treatment process. *Biotechnology & Bioengineering*, 75(4), 424–439.

Dochain, D. (2008). *Bioprocess Control*. Control Systems, Robotics and Manufacturing. Wiley.

García-Sandoval, J. (2006). *The Robust Regulation Problem Using Immersions: Reactors Applications*. Ph.D. thesis, CINVESTAV.

Henze, M. and Harremoes, P. (1983). Anaerobic treatment of wastewater in fixed film reactors- a literature review. *Water Science and Technology*, 15(1), 1–101.

Hess, J. and Bernard, O. (2008). Design and study of a risk management criterion for an unstable anaerobic wastewater treatment process. *Journal of Process Control*, 18(1), 71–79.

Huntington, R. (1998). Twenty years development of ICA in a water utility. *Wat. Sci. Technol.*, 37(12), 27–34.

Kailath, T. (1980). *Linear Systems*. Prentice Hall.

Liu, J., Muñoz de la Peña, D., Ohran, B.J., Christofides, P.D., and Davis, J.F. (2008). A two-tier architecture for networked process control. *Chemical Engineering Science*, (63), 5394–5409.

Meleiro, L. and Filho, R. (2000). State and parameter estimation based on a nonlinear filter applied t an industrial process control of ethanol production. *Braz. J. Chem. Eng.*, 17, 4–7.

Méndez-Acosta, H., Palacios-Ruiz, B., Alcaraz-González, V., Steyer, J., González-Álvarez, V., and Latrille, E. (2008). Robust control of volatile fatty acids in a anaerobic digesters. *Industrial and Engineering Chemical Research*, 47(20), 7715–7720.

Muñoz de la Peña, D. and Christofides, P.D. (2008). Output feedback control of nonlinear systems subject to sensor data losses. *Science Direct*, (57), 631–642.
Nguang, S. and Shi, P. (2003). Fuzzy $\mathcal{H}_\infty$ output feedback control of nonlinear systems under sampled measurements. *Automatica*, 39, 2169–2174.
Ray, W. (1980). *Advanced Process Control*. McGraw-Hill.
Scali, C., Morretta, M., and Semino, D. (1997). Control of the quality of polymer products on continuous reactors: Comparison of performance of state estimators with and without updating of parameters. *Journal of Process Control*, 7(5), 357–369.

Appendix A. APPENDIX

Proof. [Lemma 1] Let us define

$$\xi(t) = x(t) - z(t), \quad \text{and } \xi\left(k\delta^+\right) = x(k\delta) - z\left(k\delta^+\right),$$

where $\xi(t)$ represents the continuous error and $\xi\left(k\delta^+\right)$ is the updated error for each sampling period. Note that $x\left(k\delta^+\right) = x(k\delta)$ since system (1) is continuous. Now

$$\dot{\xi}(t) = A\xi(t) \qquad \forall t \neq k\delta \tag{A.1}$$

$$\xi\left(k\delta^+\right) = (I + GC)\,\xi(k\delta) \qquad t = k\delta. \tag{A.2}$$

Solving (A.1) for $t \in \left[k\delta^+, (k+1)\,\delta\right]$, it follows that

$$\xi(k+1) = A_d\xi\left(k\delta^+\right), \tag{A.3}$$

where $A_d = e^{A\delta}$. From (A.2) and (A.3) it is obtained

$$\begin{aligned}\xi\left((k+1)\,\delta^+\right) &= (I + GC)\,\xi(k+1)\\ &= (I + GC)\,A_d\xi\left(k\delta^+\right),\end{aligned}$$

and thus, if the pair (A_d, CA_d) is observable, then a matrix G can be calculated such that $A_d + GCA_d$ is Schur and the error $\xi\left(k\delta^+\right)$ will converge to zero, hence $\lim_{k\to\infty}\left[x(k\delta) - z\left(k\delta^+\right)\right] = 0$; then for $k\delta < t \leq (k+1)\,\delta$ the solution $z(t)$ converges to $x(t)$, that is $\lim\limits_{t\to\infty}\left[x(t) - z(t)\right] = 0$. On the other hand, to prove that the pair (A_d, CA_d) is observable if the pair (A_d, C) is observable, consider its observability matrix

$$\mathcal{O} = \begin{pmatrix} CA_d \\ CA_d^2 \\ \vdots \\ CA_d^n \end{pmatrix},$$

where $A_d \in \mathbb{R}^{n\times n}$, then using the Hamilton-Cailey theorem (Kailath, 1980)

$$A_d^n = a_0 I + a_1 A_d + \cdots + a_{n-1}A_d^{n-1},$$

the observability matrix becomes

$$\mathcal{O} = \begin{pmatrix} CA_d \\ CA_d^2 \\ \vdots \\ a_0 C + a_1 CA_d + \cdots + a_{n-1}CA_d^{n-1} \end{pmatrix}.$$

Since A_d is obtained through a discretization of matrix A then $a_0 \neq 0$ and $\mathcal{O}$ has full rank if the pair (A_d, C) is observable.

Soft sensing for two-phase flow using an ensemble Kalman filter

A. Gryzlov*, M. Leskens, R.F. Mudde***

** Department of Multi-Scale Physics, Delft University of Technology, Delft, 2628 BW, the Netherlands (Tel: 31(0)152783210; e-mail: a.gryzlov@tudelft.nl)*
*** Department of Process Modelling and Control, TNO Science and Industry, Eindhoven, the Netherlands (e-mail: martijn.leskens@tno.nl)*

Abstract: A new approach for real-time monitoring of horizontal wells, which is based on data assimilation concepts, is presented. Such methodology can be used when the direct measurement of multiphase flow rates is unfeasible or even unavailable. The real-time estimator proposed is an ensemble Kalman filter employing a dynamic model of the pipe flow and information from several downhole pressure sensors with a single measurement of the flow velocity and composition. By means of simulation examples it is shown that the proposed algorithm operates quite accurately both for noisy synthetic measurements and artificial data generated by the OLGA simulator.

Keywords: Distributed state estimation, Ensemble Kalman filter, Two-phase flow, Inverse dynamic problem

1. INTRODUCTION

The growing demand for hydrocarbon production has resulted into improved oilfield management with various monitoring and optimization strategies (Glandt, 2003, Jansen *et al.*, 2008). These strategies in turn strongly rely on the efficiency of downhole equipment which is used to obtain real-time oil and gas production rates with sufficient spatial and temporal resolution. In particular, multiphase flowmeters installed downhole can improve the production of long horizontal wells by allocating the zones of oil, gas and water inflow. However, existing multiphase meters are expensive, inaccurate or accurate only within a limited operating range and therefore such monitoring is unrealistic.

To overcome these problems one can use so-called multiphase soft-sensors, i.e. to estimate flow rates from conventional meters, such as downhole pressure gauges, in combination with a dynamic multiphase flow model. Despite the variety of soft-sensing techniques (which are also referred to as data assimilation methods), one can note two principal approaches. Variational data assimilation, which is based on the minimization of a cost function within a certain time interval, and sequential methods or filtering when the state of the system is updated every time instant data becomes available. One way to solve these sequential data assimilation problems is to use Kalman filtering (Kalman, 1960). This method, which was originally developed for linear models, has got numerous extensions (Jazwinski, 1970, Evensen, 1994 and Julier *et al.*, 2000) to deal with non-linearity, which is the case for most industrial processes.

Although there are numerous applications of soft-sensing techniques in oil and gas industry, they mainly deal with the estimation of reservoir properties (Naevdal *et al.*, 2003, Evensen *et al.*, 2007). The range of wellbore flow application includes gas-lift wells (Bloemen *et al.*, 2004) and underbalanced drilling (Lorentzen *et al.*, 2001). Also, the Kalman filter has been used for tuning the parameters of two phase flow models (Lorentzen *et al.*, 2003). Leskens *et al.* (2008) considered the simultaneous estimation of downhole oil, water and gas flow rates from downhole pressure and temperature measurements in a single well. This approach has been extended by de Kruif *et al.* (2008) to the multi-lateral well case both for the two-phase (oil and gas) and three-phase (oil, gas and water) cases.

Despite the variety of applications considered, little attention has been given to the inflow allocation problem. More specific, long horizontal wells with a continuous inflow profile from a reservoir to a wellbore require the use of soft-sensing techniques for the gas breakthrough prediction.

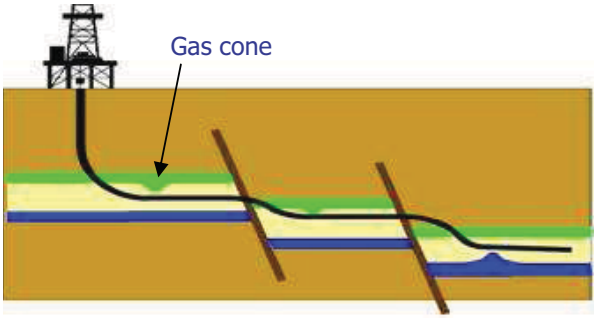


Fig. 1. Schematic view of a horizontal well.

Gas coning is a phenomenon where the gas-oil contact of a reservoir moves towards a producing well (see Figure 1). At a certain moment the gas-oil contact will reach the well and gas breakthrough can happen causing a large gas influx. Consequently, the gas phase may start to dominate production making the well uneconomical. In order to handle or prevent this, several strategies are available. However, the most convenient countermeasure is to isolate gas producing

zones of a wellbore by means of inflow control valves. The purpose of a soft-sensor is to provide the inflow control valves with information of the downhole flow rate distribution.

This study discusses the feasibility of such multiphase soft-sensors. In particular, the required and sufficient set of measurements is defined. Furthermore, the influence of model error and measurement noise on the quality of estimates is studied.

This paper is organized as follows. First, the pipe flow model and the computational setup for soft-sensing are given. Next, the description of the used soft-sensing algorithm is presented. Finally, the simulations results are given.

2. DYNAMIC FLOW MODEL

A model describing one-dimensional two-phase flow in pipes consists of non-linear partial differential equations describing mass and momentum conservation for each phase. This model is obtained from cross-sectional averaging of the Navier-Stokes equations and replacing diffusion terms by empirical correlations. Since the main purpose of this work is the application of estimation techniques, no detailed flow description is required. Therefore, it was assumed that the gas and liquid are travelling with the same velocity u (Vicente, *et al.*, 2001).

The simplified mass conservation equations are

$$\frac{\partial}{\partial t}(\rho_l H) + \frac{\partial}{\partial s}(\rho_l H u) = \Phi_l \tag{1}$$

$$\frac{\partial}{\partial t}(\rho_g (1-H)) + \frac{\partial}{\partial s}(\rho_g (1-H) u) = \Phi_g \tag{2}$$

Where H is the liquid volume fraction, ρ_g is the gas density, ρ_l is the liquid density, t denotes time and s denotes the coordinate along the length of the pipe. Φ_l and Φ_g are the mass sources representing the inflow from a reservoir to the pipe. These sources are normally time dependent.

Although the continuity equations have been written for each phase it is common to write the momentum equation for the mixture.

$$\frac{\partial}{\partial t}(\rho_m u) + \frac{\partial}{\partial s}(\rho_m u^2) = -\frac{\partial p}{\partial s} - S_{fr} \tag{3}$$

Where ρ_m is the mixture density defined by

$$\rho_m = \rho_g (1-H) + \rho_l H \tag{4}$$

A frequently used model for frictional losses in the momentum equation has the form

$$S_{fr} = \frac{\lambda}{2d} \rho_m u^2 \tag{5}$$

Here d is the pipe diameter and λ is the friction factor, which is a function of the Reynolds number and pipe roughness k. In this study the Techo formula is used:

$$\lambda = \left[-0.8685 \ln\left(\frac{1.964 \ln(\mathrm{Re}) - 3.8215}{\mathrm{Re}} + \frac{k}{d \cdot 3.71} \right) \right]^{-2} \tag{6}$$

Here Re is the Reynolds number defined as

$$\mathrm{Re} = u d \rho_m / \mu_m \tag{7}$$

with the mixture viscosity μ_m calculated in terms of liquid volume fraction and gas μ_g and liquid μ_l viscosities

$$\mu_m = \mu_g (1-H) + \mu_l H \tag{8}$$

The gas is treated as a compressible phase with a corresponding equation of state given in the form

$$\rho_g = f(p) \tag{9}$$

The closure of the problem is given by the following boundary conditions.

$$p(s=L,t) = p_{out}, \; p(s,t=0) = p_{out} \tag{10}$$

$$u(s=0,t) = u_{\inf}, \; u(s,t=0) = u_{\inf} \tag{11}$$

$$H(s=0,t) = H_{\inf}, \; H(s,t=0) = H_{\inf} \tag{12}$$

Here the subscripts *inf* and *out* refer to inflow and outflow cross-section of the pipe respectively. L denotes the length of the pipe.

3. DATA ASSIMILATION

3.1 State-space form of the model equations

Due to the nonlinearity of the given equation system (1)-(12) the numerical solution is needed in order to solve it for the dependent variables. For the discretization of the simulation domain a staggered grid approach has been used, meaning that the different grids are used for the continuity and momentum equation. Afterwards, the governing equations are integrated over different control volumes. Any solution procedure can be applied for solving the non-linear system of algebraic equations. Finally, the model can be written in the following state-space notation (Crassidis, 2004):

$$x_k = f(x_{k-1}, u_{k-1}) \tag{13}$$

Here u_{k-1} is the model input representing the inflow from reservoir to wellbore. x_{k-1} is the state vector evaluated on the previous time step. Using the primitive set of variables, the state vector can be written as

$$x = [\, p \; u \; H \,]^T \tag{14}$$

Here p, u and H are the vectors, representing pressure, velocity and liquid volume fraction related to the spatial grid.

3.2 Formulation of the inverse problem

The computational setup for the inverse problem is shown in Figure 2. It should be noted here, that only the horizontal part of the well is being modelled, and the outflow measurements are assumed to be available directly at the outflow cross-section of the horizontal part.

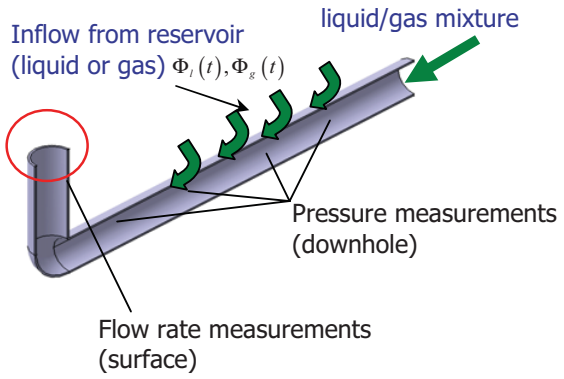


Fig. 2. Scheme of the computational setup for soft-sensing.

For the soft sensing purposes the augmented state vector is introduced:

$$X = \left[p_i \; u_i \; H_i \left| \Phi_{g\,i} \; \Phi_{l\,i} \right. \right]^T \tag{15}$$

Here i indicates the number of the cell defined by the numerical discretization.

It is assumed that several downhole pressure measurements are available. Moreover, outflow information about flow rates is also known, giving the following measurement vector:

$$y = \left[p_i \; u_{out} \; H_{out} \right]^T \tag{16}$$

Finally, the data assimilation problem can be formulated as follows: with the measurements (16) and the flow model (1)-(12) available the components of the augmented state vector should be estimated.

Due to a lack of experimental data, a set of synthetic measurements has been used as a source for soft-sensing. First, a twin experiment concept has been implemented. Here the same mathematical model was used both for generating measurements with predefined inflow distribution and the inverse modelling, when missing dynamic variables are estimated by means of the soft-sensing algorithm. In order to mimic the situation of testing the soft-sensor with "real-life" data, simulation results from the commercially available flow simulator OLGA were used in the second test case.

3.3 Ensemble Kalman filtering

One way to solve estimation problems via the sequential data assimilation algorithm is by using the Kalman filter equations. The Kalman filter is a stochastic recursive estimator, which estimates the values of model states and unknown input by integrating measured data in a mathematical model in real-time. Due to its straightforward numerical implementation and recursive nature, the Kalman filter algorithm is very well adapted to online model calibration.

Kalman filtering was initially developed for linear dynamic systems. Although several extensions of the Kalman filter exist for non-linear system, here the ensemble Kalman filter (EnKF) is used (Evensen, 1994). In this approach, the approximation of the error covariance matrix is calculated using an ensemble of possible model realizations, which are propagated according to the full dynamics of the system.

In order to initialize the filter the initial ensemble is generated. Here a mean value of the initial augmented state vector $\overline{X}_0^a$ and a corresponding covariance matrix Q_o is required. The mean value of the initial ensemble should be a good estimate of the true initial state. The members of the ensemble are generated randomly according to a Gaussian distribution. The *j'*th member of the ensemble is defined as

$$X_{0,j}^a = \overline{X}_0^a + w_{0,j} \tag{17}$$

With an EnKF the augmented state vector, which also contains the inflow input, is estimated in a recursive manner through the following two steps:

1) The forecast step, which consists in running the flow model one time step forward for each member of the ensemble. This leads to

$$X_{k,j}^f = f\left(X_{k-1,j}^a \right) + w_{k,j} \tag{18}$$

Here $w_{k,j}$ is a Gaussian zero mean white noise with the corresponding covariance matrix Q_k representing the model error. This noise is only added to components of the state vector, which produce the most uncertainty in a simulation. These are in this case the inflow sources Φ_l and Φ_g.

Using the calculated forecast of ensemble states, the error covariance matrix can be calculated using the covariance matrix of the ensemble. The mean value of the ensemble is given by

$$\overline{X}_k^f = \frac{1}{N} \sum_{j=1}^{N} X_{k,j}^f \tag{19}$$

And the error covariance matrix is then calculated as

$$P_k^f = L_k^f (L_k^f)^T \tag{20}$$

With

$$L_k^f = \frac{1}{\sqrt{N-1}} [(X_{k,1}^f - \overline{X}_k^f) \; (X_{k,2}^f - \overline{X}_k^f) ... (X_{k,N}^f - \overline{X}_k^f)] \tag{21}$$

where N is the number of members in the ensemble.

2) The analysis step, which takes into account measurements. The errors in the measurements are assumed to be statistically independent with known variances. This leads to a diagonal covariance matrix for the measurement errors. As it has been pointed out in Burgers *et al.* (1998), it is necessary to define new measurements for the proper error propagation. Therefore, a new observation vector is introduced for each member of the ensemble

$$y_{k,j} = M_k \cdot X_{k,j} + v_{k,j} \tag{22}$$

Here M_k is the measurement matrix and $v_{k,j}$ is the measurement noise generated from a normal distribution with zero mean and covariance matrix R_k.

The Kalman gain is then calculated as follows

$$K_k = P_k^f M_k^T (M_k P_k^f M_k^T + R_k)^{-1} \quad (23)$$

The analyzed state for each member of the ensemble is given by

$$X_{k,j}^a = X_{k,j}^f + K_k (y_{k,j} - M_k X_{k,j}^f) \quad (24)$$

The mean value of the analyzed ensemble is

$$\overline{X}_k^a = \frac{1}{N} \sum_{j=1}^{N} X_{k,j}^a \quad (25)$$

The unknown inflow sources are updated at each time step measurements are available and extracted from the augmented state vector. The analyzed error covariance matrix, from which the estimation error of the inflow parameters can be defined, is then approximated by

$$P_k^a = (I - K_k M_k) P_k^f \quad (26)$$

An important issue with the use of the EnKF is the size of the ensemble. Based on the experience of data assimilation for large-scale atmospheric models (Houtekamer and Mitchell, 1998), 100 ensemble members have been chosen for the ensemble Kalman filtering. The optimal size of the ensemble is, however, not known and it is a subject for future research.

4. RESULTS AND DISCUSSIONS

4.1 Soft sensing under measurement error

A first test case considered uses a twin experiment concept. Here the same mathematical model is used for generating the measurements with predefined inflow distribution. This study deals with two-phase liquid/gas flow and the details of the initial data are given in Table 1. The sketch of the simulation domain is given in Figure 3. The inflow profiles are given only as a reference since they are unknown and have to be estimated via the proposed data assimilation procedure.

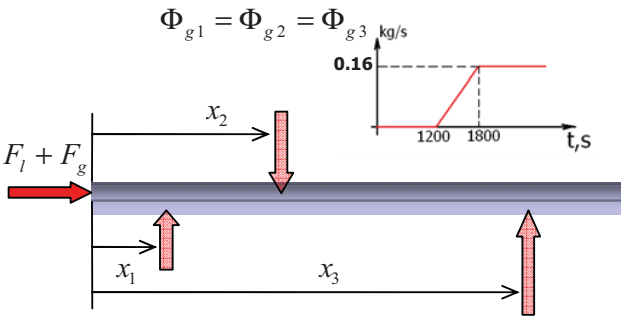


Fig. 3. Computational setup for soft-sensing.

Initially the well produces a mixture of liquid and gas with a total flow rate of 10 kg/s. After 20 minutes of production, gas is injected in three locations of the wellbore. The amount of gas injected increases linearly up to 0.5 kg/s during next 30 minutes and afterwards kept constant for the last 10 minutes of simulation.

The soft-sensor has been tested using the following measurement layout. The number of pressure measurements was taken equal to number of grid nodes obtained from the discretization. The velocity and liquid volume fraction measurements are located at the last grid block of the simulation domain.

Table 1. Initial data for the numerical experiments

Quantity	Value
Pipe diameter, m	0.05
Pipe length, m	100
Liquid density, kg/m^3	1000
Liquid viscosity, Pa·s	0.001
Gas reference density, kg/m^3	118.9
Gas viscosity, Pa·s	$1.82 \cdot 10^{-5}$
Time step, s	60
Inflow liquid rate F_l, kg/s	9.5
Inflow gas rate F_g, kg/s	0.5
x_1, m	15
x_2, m	45
x_3, m	75
Absolute roughness, m	0
Number of grid nodes	12

The Kalman filter initialization is based here on the outflow values of velocity and liquid volume fraction, which are assumed to be known from a flow meter. Since all the pressure measurements are available, pressure is initialized from the current pressure distribution. The synthetic measurements representing downhole pressure and liquid outflow flow rate are generated using equations (1)-(12). A zero mean white Gaussian noise is then added to mimic the uncertainty in measurements.

Table 2. Measurement noise used in simulations

Uncertainty in pressure measurements	0.5%
Uncertainty in outflow velocity measurements	1%
Uncertainty in liquid volume fraction measurements	1%

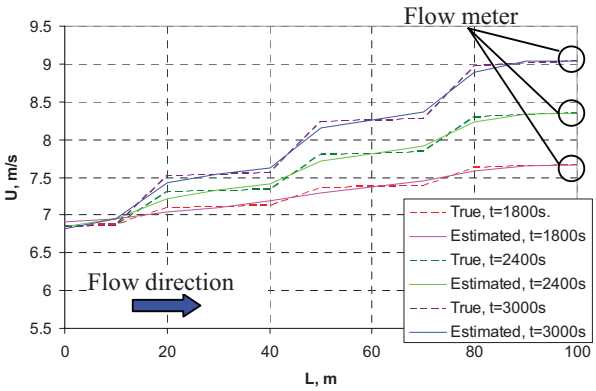


Fig. 4. Comparison of estimated and true flow velocity.

The results of the simulation are given in Figures 4-5. Figure 4 shows the comparison between the estimated and true velocity distributions along the pipe length. Flow velocity is used to allocate the zones where a fluid is entering or leaving the wellbore. In order to identify the type of fluid, the distribution of the estimated liquid volume fraction is required. It is depicted in Figure 5. The results are given for three time instants 30 minutes, 40 minutes and 50 minutes. Since the pressure is available continuously from the measurements it is not depicted as a soft-sensing result.

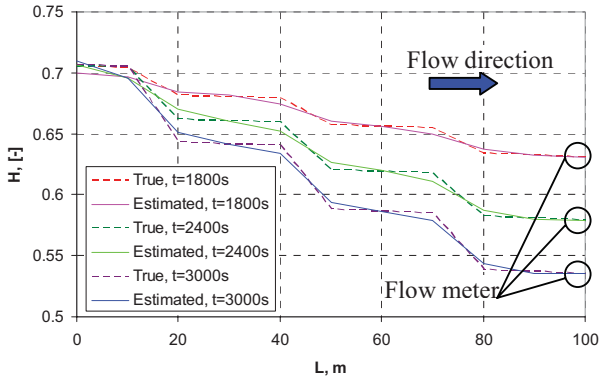


Fig. 5. Comparison of estimated and true liquid fraction.

The results show that the proposed soft-sensor, for the given simplified setup, is very well capable of reproducing the flow rate and liquid volume fraction distributions along the considered well part, even when measured data contains a certain measurement error. Therefore, it is capable to detect multiple fluid sources as it is depicted in the figures.

4.2 Soft sensing under model error

The second study provides an assessment of the influence of the model error on the soft-sensing estimation results. A similar soft-sensing setup was used as depicted in Figure 2 for case study 1. An important difference, however, was that the "true" well was not the same as the model used in the soft-sensor. The true wellbore measurements were obtained from the commercially available simulator OLGA. This was done to assess the inevitable effect of the model error on the soft-sensing estimation results. Here both transient gas and liquid sources are present in a computational setup. Liquid is injected in the first part of the pipe, while a gas source is present close to its outflow cross-section. This situation is a rough approximation of the gas breakthrough scenario. The scheme of the simulation domain is given in Figure 6.

Due to differences between the flow model used in OLGA simulator and the soft-sensor developed, one can point at the following sources of the model error:

- Friction factor correlation

- Fluid properties

- Simulation grid

- Mathematical model

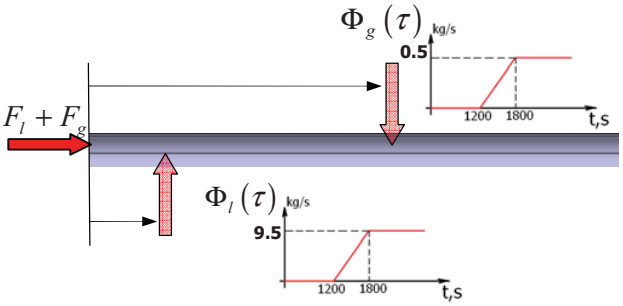


Fig. 6. Computational setup for soft-sensing. Test case 2.

A particularly important modelling assumption for performing OLGA simulations was to keep a dispersed bubble flow regime, since the model used for soft-sensing is valid only for that type of multiphase flow. This was possible using the same set of input parameters, as for the test case 1. The OLGA simulations were performed with 10 grid nodes, where the source term for liquid has been defined in the third grid block, and for gas in the eighth grid block. This consequently led to a soft-sensing setup with 10 available pressure measurements.

Figures 7 and 8 represent the estimated flow velocity and liquid volume fraction.

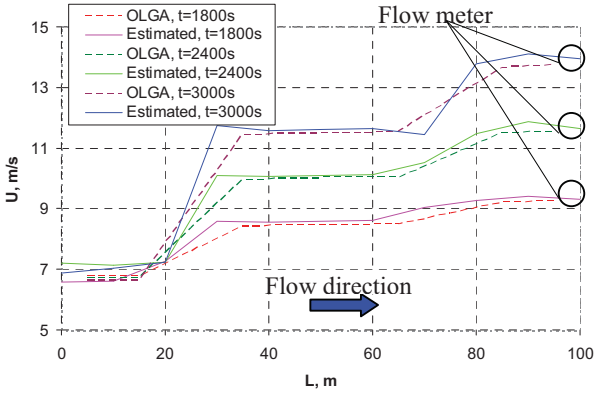


Fig. 7. Estimated velocity profile for the OLGA data.

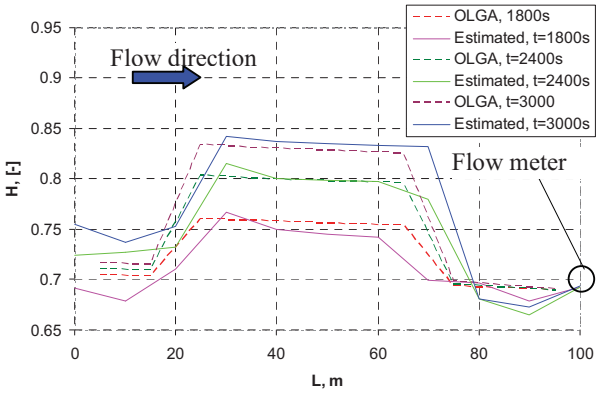


Fig. 8. Estimated liquid fraction profile for the OLGA data.

The results obtained under model error are not as accurate as for the twin-experiment. However, it is still possible to allocate easily zones of liquid or gas inflow. A displacement of the estimated profiles with respect to the true ones is observed. This can be explained by the use of a different grid in the OLGA simulator and different interpolation of the flow variables between grid nodes and edges.

5. SUMMARY AND CONCLUSIONS

By means of two case studies, some limitations and possibilities of soft-sensor multiphase flow meters have been studied. The proposed soft-sensor is based on the ensemble Kalman filter approach and requires as the input the dynamic model of the pipe flow together with pressure measurements available downhole and one composition and velocity measurement at the outflow.

It has been shown, that for a two-phase flow formulation it is possible to reconstruct the distributions of the flow velocity and liquid volume fraction along a pipe and to allocate the inflow of certain fluids in a specific location along it.

The results indicate that the proposed method is quite stable for a certain range of wellbore operational conditions, and capable of taking into account measurement and model error.

6. ACKNOWLEDGEMENTS

This work has been supported by ISAPP knowledge center, which is a joint research project of Shell, TNO and Delft University of Technology. The authors would also like to thank Wouter Schiferli (TNO Science and Industry) for performing the OLGA runs.

REFERENCES

Bloemen, H.H.J., Belfroid, S.P.C., Sturm, W.L., and Verhelst, F.J.P.C.M.G. (2004). Soft Sensing for Gas-Lift Wells. SPE paper 90370, *Proc. SPE Annual Technical Conference and Exhibition*, Houston, Texas, USA.

Burgers, G., van Leeuwen, P.J., and Evensen, G. (1998) On the analysis scheme in the ensemble Kalman Filter. *Monthly Weather review*. Vol. 126, 1719-1734.

Crassidis J.L. and Junkins J.L. (2004). *Optimal estimation of Dynamic systems*. Chapman & Hall/CRC. Florida.

De Kruif, B., Leskens, M., van der Linden, R., Alberts, G., and Smeulers, J. (2008) Soft-sensing for multilateral wells with downhole pressure and temperature measurements. SPE paper 118171, *Proc. Dhabi International Petroleum Exhibition and Conference*, Abu Dhabi, UAE,

Evensen, G. (1994). Sequential data assimilation with a nonlinear quasi-geostrophic model using Monte Carlo methods to forecast error statistics," *J. Geophys. Res*., Vol. 99(C5), 10143–10162.

Evensen, G., Hove J., Meisingset, H.C., Reiso E., Seim, K.S., and Espelid, O. (2007). Using the EnKF for Assisted History Matching of a North Sea Reservoir Model. SPE paper 106184, *Proc. SPE Reservoir simulation symposium*, Woodlands, Texas, USA.

Glandt, C.A. (2003). Reservoir aspects of smart wells. SPE paper 81107, *Proc. SPE Latin American and Caribbean Petroleum Engineering Conference*, Trinidad, West Indies.

Houtekamer, P.L., Mitchell, H.L., (1998). Data assimilation using an ensemble Kalman filter technique. *Monthly Weather review*. Vol. 126, 796-911.

Jansen J.D., Bosgra O.H., and Van den Hof P.M.J. (2008). Model-based control of multiphase flow in subsurface oil reservoirs. *Journal of Process Control*, Vol. 18, 846-855.

Jazwinski, A. H. (1970). *Stochastic Processes and Filtering Theory*. Academic Press, New York.

Julier, S., Uhlmann, J., and Durrant-Whyte, H.F. (2000). A new method for the non-linear transformation of means and covariances in filters and estimators. *IEEE Transcations on Automatic Control*. Vol. 45(3), pp. 477-482.

Kalman, R.E. (1960). A new approach to linear filter and prediction theory. *Trans. ASME, Series D, Journal of Basic Engineering,* Vol. 82, pp. 35 – 45.

Leskens, M., de Kruif, B., Belfroid, S., Gryzlov, A., and Smeulers, J. (2008). Downhole Multiphase Metering in Wells by Means of Soft-Sensing. SPE paper 112046, *proc. SPE Intelligent Energy Conference*, Amsterdam, The Netherlands.

Lorentzen, R.J., Fjelde, K.K.,Frøyen J, Lage A. C. V. M., Nævdal G., and Vefring E.H. (2001). Underbalanced and Low-head Drilling Operations: Real Time Interpretation of Measured Data and Operational Support, *Proc. SPE Annual Technical Conference and Exhibition*, SPE 71384, New Orleans, Louisiana.

Lorentzen, R.J., Nævdal, G., and Lage, A.C.V.M. (2003). Tuning of parameters in a two-phase flow model using an ensemble Kalman filter. *International Journal of Multiphase Flow*. Vol. 29(8), pp. 1283-1309.

Nævdal, G., Johnsen, L.M., Aanonsen, S.I., and Vefring, E.H. (2003). Reservoir Monitoring and Continuous Model Updating Using Ensemble Kalman Filter. SPE paper 84372, *Proc. SPE Annual Technical Conference and Exhibition,* Denver, Colorado, USA.

Vicente, R., Sarica, C., and Ertekin, T. (2001) .A Two-Phase Model Coupling Reservoir and Horizontal Well Flow Dynamics. SPE paper 69570, *Proc. SPE Latin American and Caribbean Petroleum Engineering Conference*, Buenos Aires, Argentina.

Efficient Moving Horizon State and Parameter Estimation for the Varicol SMB Process

Achim Küpper [*], **Moritz Diehl** [***], **Johannes P. Schlöder** [**], **Hans Georg Bock** [**], **Sebastian Engell** [*]

[*] *Process Dynamics and Operations Group, Technische Universität Dortmund, Emil-Figge-Str. 70, 44221 Dortmund, Germany (e-mail: achim.kuepper@bci.tu-dortmund.de).*
[**] *IWR - Interdisciplinary Center for Scientific Computing, Universität Heidelberg, Germany.*
[***] *Electrical Engineering Department (ESAT-SCD), K.U. Leuven, Belgium.*

Abstract: In this paper, a moving horizon state and parameter estimation (MHE) scheme for the Varicol process is presented. The Varicol process is an extension of the Simulated Moving Bed (SMB) process that realizes non-integer column distributions over the separation zones by an asynchronous switching of the inlet and outlet ports (the ports are shifted individually). These additional degrees of freedom can be used to yield an improvement in economical performance compared to SMB operation. The proposed estimation scheme is based on a rigorous SMB model that incorporates rigorous chromatographic columns and port switching. The absence of model simplifications allows the extension of the estimation scheme to the more complex Varicol process. The goal of the estimation scheme is to reconstruct the full state of the system, i.e. the concentration profiles along all columns, and to identify critical model parameters in the presence of noisy measurements. The estimation is based on measurements of the concentrations of the components at the two outlet ports (which are asynchronously switched from one column to the next) and at one fixed location between two columns. The state estimation scheme utilizes a deterministic model within the prediction horizon. State noise is only considered in the state and in the parameters up to the beginning of the horizon. By applying a multiple-shooting method and a real-time iteration scheme for solving the resulting optimization problem, the computation times are reduced and the scheme can be applied online. A numerical simulation for an enantiomer separation system with nonlinear adsorption isotherm is presented.

Keywords: Varicol, Simulated Moving Bed chromatography, moving horizon estimation, state estimation, model identification, real-time application, real-time iteration

1. INTRODUCTION

The Simulated Moving Bed (SMB) process is an efficient chromatographic separation technology that is increasingly applied in the food, fine chemicals, and pharmaceutical industries. Industrial applications have been reported especially for the separation of temperature sensitive components and for the separation of species with similar thermodynamic properties. A SMB process is realized by connecting several chromatographic columns in a closed loop as illustrated by Figure 1. The Varicol process switches the ports indvidually and thereby realizes non-integer column distributions over the zones **?**), see Figure 2.

SMB processes and their variants are characterized by mixed discrete and continuous dynamics, spatially distributed state variables with steep slopes, and slow and strongly nonlinear responses of the concentrations profiles to changes of the operating parameters, therefore, they are difficult to control and to observe. In the literature, relatively few contributions that deal with state estimation of SMB processes can be found. The published work is based upon the approximation of the concentration profiles by a set of truncated exponential functions Alamir and Corriou (2003), or by using the equivalent True Moving Bed (TMB) model Mangold et al. (1994), Kloppenburg and Gilles (1999), or deals with the engineering of tailored estimation schemes Küpper and Engell (2006), Kleinert and Lunze (2005). Recently, a rigorous moving horizon estimation approach for SMB processes was proposed by Küpper et al. (2009). In this formulation of the MHE, a deterministic behaviour of the process on the estimation horizon and Gaussian independent identically distributed measurement noise are assumed. The initial state at the beginning of the horizon and its covariance are computed by an Extended Kalman Filter (EKF). The state noise covariance and the initial error covariance of this EKF are the only tuning parameters of this scheme. A fast online solution of the underlying constrained least-squares optimization problem is obtained by using the direct multiple shooting method Bock (1981, 1987). A full rigorous process

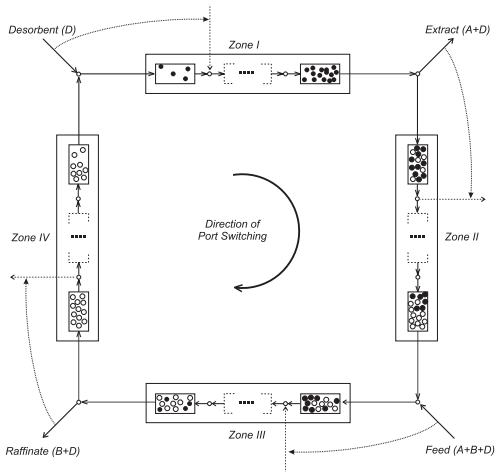


Fig. 1. Principle of the Simulated Moving Bed process

model is applied and therefore no assumption that the plant is close to the periodic steady state is needed. Along with the states, key adsorption parameters are estimated online. Simulations demonstrate that the states and critical model parameters can be reconstructed successfully. The scheme also works during transition periods including the start-up phase. The computation times are such that the estimator can be applied online. Since a rigorous full scale SMB model is used, the MHE approach can be extended to more complex variants of the SMB process. In this contribution, the moving horizon state and parameter estimation scheme is applied to the Varicol operation.

2. THE VARICOL SMB PROCESS

Chromatographic separation is based on the different adsorption affinities of the molecules in the liquid to an adsorbent which is packed in a chromatographic solid bed. The SMB process realizes a counter-current movement between the liquid and the adsorbent by switching the ports in the direction of the liquid flow periodically, as illustrated by Figure 1. In the Varicol process, the individual ports i (Eluent, Extract, Feed, Raffinate) are switched individually at the subperiod times δt_i, as illustrated by Figure 2. The individual port switching reduces the impurities by early switching of the Raffinate port and delayed switching of the Extract port. Since the Varicol process offers a larger number of degrees of freedom, it can be operated with better process economics than the SMB process, see Toumi et al. (2002), Toumi et al. (2003).

In the estimation scheme, the counter-current flow of the solid and of the liquid phases is modelled in the same way as it is achieved in the real plant by asynchronously switching the inlet and outlet ports in the direction of the liquid flow after subperiod n with subperiod length $(\delta t_n - \delta t_{n-1})\tau$ has passed. The state variables represent the concentrations in the physical columns and do not exhibit jumps. Only the input flow rates and the inflow concentrations change discontinuously. The dynamic simulation of the Varicol process is achieved by integrating the differential equation over the subperiods n

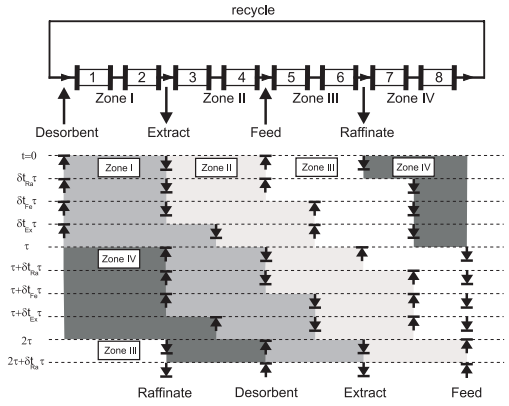


Fig. 2. Asynchronous switching in the Varicol process

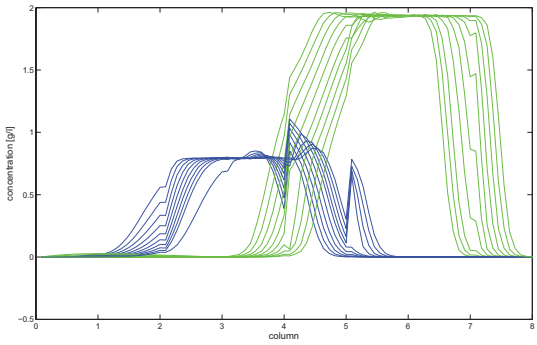


Fig. 3. Concentration profiles of the Varicol process during one period with $\delta\tau_{Ra} = 0.3$, $\delta\tau_{Fe} = 0.6$, $\delta\tau_{Ex} = 0.9$, $\delta\tau_{De} = 1.0$, $\delta\tau_{Re} = 1.0$

$$\dot{\mathbf{x}} = f(\mathbf{x}, Q_n, \mathbf{p}) \qquad (1)$$
$$t \in [(m-1)\tau + \delta t_{n-1}\tau,\ (m-1)\tau + \delta t_n \tau]$$
$$\mathbf{x}(t_0) = \mathbf{x}_{m,0} \qquad (2)$$

followed by the switching of the flows $Q_{n,j,\delta t_j}$:

$$Q_{n+1,j,\delta t_j} = M_Q Q_{n,j,\delta t_j} \quad j = \mathrm{De}, \mathrm{Ex}, \mathrm{Fe}, \mathrm{Ra}, \mathrm{Re}, \qquad (3)$$

with differential states $\mathbf{x}(t) \in \mathbb{R}^{n_x}$ and parameters $\mathbf{p} \in \mathbb{R}^{n_p}$. The vector $Q_{n,j,\delta t_j}$ defines the inlet/outlet flow of port j (desorbent, extract, feed, raffinate) and the recycle stream at the individual switching time δt_j in period m (m denotes the full period count). The components of $Q_{n,j}$ represent the flows of the ports j to the columns. M_Q is a permutation matrix that shifts the flow ports after the asynchronous switching time δt_j of port j has passed (with individual period counter n). The recycle flow that defines the total flow rate in the zone in front of the desorbent port is switched with the last port. The zone flow rates result from the port flows and the recycle flow. The concentration profiles during one switching period are illustrated by Figure 3. The asynchronous switching of the feed port and of the extract port can be clearly seen in the profiles. In this paper, three positions where the concentrations of the two substances of the mixture are measured are assumed. The measurements are installed behind the extract port, behind the raffinate port, and behind one column in the process where physically the closing of the loop is realized (six measurements total). The extract and raffinate concentration measurements

move together with the ports. More measurements are not available in production plants.

2.1 Rigorous Dynamic Modelling

From mass balances of the components around the inlet and the outlet ports, the internal flow rates and the inlet concentrations can be calculated according to:

$$\text{Desorbent node: } Q_{\mathrm{IV}} + Q_{\mathrm{De}} = Q_{\mathrm{I}} \tag{4}$$

$$c_{i,\mathrm{out,IV}} Q_{\mathrm{IV}} = c_{i,\mathrm{in,I}} Q_{\mathrm{I}} \quad i = \mathrm{A,B} \tag{5}$$

$$\text{Extract node: } Q_{\mathrm{I}} - Q_{\mathrm{Ex}} = Q_{\mathrm{II}} \tag{6}$$

$$\text{Feed node: } Q_{\mathrm{II}} + Q_{\mathrm{Fe}} = Q_{\mathrm{III}} \quad i = \mathrm{A,B} \tag{7}$$

$$c_{i,\mathrm{out,II}} Q_{\mathrm{II}} + C_{i,\mathrm{Fe}} Q_{\mathrm{Fe}} = c_{i,\mathrm{in,III}} Q_{\mathrm{III}} \tag{8}$$

$$\text{Raffinate node: } Q_{\mathrm{Ra}} + Q_{\mathrm{IV}} = Q_{\mathrm{III}}, \tag{9}$$

where $Q_{\mathrm{I-IV}}$ are the flow rates in the corresponding zones, Q_{De}, Q_{Ex}, Q_{Fe}, and Q_{Ra} denote the external flow rates and $c_{i,\mathrm{in}}$ and $c_{i,\mathrm{out}}$ denote the concentrations of the component i in the streams leaving and entering the respective zone. The initial distribution of the columns over the four separation zones is 2/2/2/2 and the individual switching times are $\delta\tau_{Ra} = 0.3$, $\delta\tau_{Fe} = 0.6$, $\delta\tau_{Ex} = 0.9$, $\delta\tau_{De} = 1.0$, $\delta\tau_{Re} = 1.0$ from which the non-integer column distribution 2.1/2.3/2.3/1.3 results.

The chromatographic columns are modelled by the General Rate Model. It is assumed that there are no radial gradients in the column and that the particles of the solid phase are uniform, spherical, porous (with a constant particle porosity ϵ_{p}), and that the mass transfer between the particle and the surrounding layer of the bulk is in local equilibrium. The concentration of component i is denoted by c_i in the liquid phase and by q_i in the solid phase. D_{ax} is the axial dispersion coefficient, u the interstitial velocity, ϵ_{b} the void fraction of the bulk phase, $k_{\mathrm{l},i}$ the film mass transfer resistance, and D_{p} the diffusion coefficient within the particle pores. The concentration within the pores is denoted by $c_{\mathrm{p},i}$. The following partial differential equations of a column can be derived from a mass balance around an infinitely small cross section area of the column assuming a constant radial distribution of the interstitial velocity u and the concentration c_i.

$$\frac{\partial c_i}{\partial t} + \frac{(1-\epsilon_{\mathrm{b}}) 3 k_{\mathrm{l},i}}{\epsilon_{\mathrm{b}} r_{\mathrm{p}}} \left(c_i - c_{\mathrm{p},i}|_{r=r_{\mathrm{p}}}\right) = D_{\mathrm{ax}} \frac{\partial^2 c_i}{\partial x^2} - u \frac{\partial c_i}{\partial x} \tag{10}$$

$$(1-\epsilon_{\mathrm{p}}) \frac{\partial q_i}{\partial t} + \epsilon_{\mathrm{p}} \frac{\partial c_{\mathrm{p},i}}{\partial t} - \epsilon_{\mathrm{p}} D_{\mathrm{p}} \left[\frac{1}{r^2} \frac{\partial}{\partial r} \left(r^2 \frac{\partial c_{\mathrm{p},i}}{\partial r}\right)\right] = 0, \tag{11}$$

with appropriate initial and boundary conditions. It is assumed that the concentration q_i is in thermodynamic equilibrium with the liquid concentrations in the particle and their relationship can be described by an extended Langmuir adsorption isotherm

$$q_i = H_i^1 c_{\mathrm{p},i} + \frac{H_i^2 c_{\mathrm{p},i}}{1 + k_{\mathrm{A}} c_{\mathrm{p,A}} + k_{\mathrm{B}} c_{\mathrm{p,B}}} \quad i = \mathrm{A,B}, \tag{12}$$

with H_i^j and k_i as isotherm constants. The resulting system of coupled differential equations can be efficiently solved by the numerical approach proposed in Gu (1995) where a Galerkin finite element discretization of the bulk phase is combined with orthogonal collocation for the solid phase. This numerical method was first applied to SMB processes in Dünnebier and Klatt (2000). The bulk phase is divided into n_{fe} finite elements and the solid phase is discretized by n_{c} internal collocation points. As a result, the set of initial values, boundary values, and partial differential equations (PDE) is transformed into a set of initial values and a system of ordinary differential equations (ODE)

$$\dot{\mathbf{x}} = f(\mathbf{x}, \mathbf{u}, \mathbf{p}), \tag{13}$$

where the flows Q are summarized in the input vector $\mathbf{u}(t) \in \mathbb{R}^{n_u}$. The system output is defined as

$$y = h(\mathbf{x}(\mathbf{u}, \mathbf{p})), \tag{14}$$

with $y \in \mathbb{R}^{n_y}$. For $n_{\mathrm{fe}} = 5$, $n_{\mathrm{c}} = 1$, number of components $n_{\mathrm{sp}} = 2$, and number of columns $n_{\mathrm{col}} = 8$ a system order of the SMB process of

$$n_x = n_{\mathrm{col}} * n_{\mathrm{sp}} * (n_{\mathrm{c}} + 1) * (2 * n_{\mathrm{fe}} + 1) = 352 \tag{15}$$

results. The ODE-system is stiff due to large differences in the speeds of the interacting dynamics.

3. ESTIMATION IN THE VARICOL PROCESS

3.1 Moving Horizon Estimation

For the simultaneous estimation of the states and the parameters of SMB processes, we employ the Moving Horizon Estimation scheme introduced by Diehl et al. (2006); Kühl et al. (2008), which is modified in order to handle the shift of the inputs and of the measurements of the SMB process. The Moving Horizon Estimator estimates the states and the parameters based on the past measurements at specific time points that are located in the horizon $T_N = t_K - t_L$. t_K represents the current time and t_L is the time at the beginning of the horizon. A least-squares minimization is performed that minimizes the deviations of the real measurements η_k from the simulated measurements $h(\mathbf{x}(t_k; \mathbf{u}, \mathbf{p}))$ at times t_k. The expression $||.||^2_{V^{-\frac{1}{2}}}$ denotes $||\mathbf{x}||^2_{V^{-\frac{1}{2}}} = \mathbf{x}^T V^{-1} \mathbf{x}$, where the matrix V is the positive semidefinite noise covariance matrix of the variables $\mathbf{x}$. $V^{-\frac{1}{2}}$ can be interpreted as weighting matrix of $\mathbf{x}$. The measurement information prior to the moving horizon is considered in the estimation problem by an arrival cost term that is computed from the expected value of the state and the parameters and the estimation error covariance before the horizon. The optimization problem of the MHE results as:

$$\min_{\mathbf{x}(t_L),p} \left\{ \left\| \begin{matrix} \mathbf{x}(t_L) - \overline{\mathbf{x}}_L \\ \mathbf{p} - \overline{\mathbf{p}}_L \end{matrix} \right\|^2_{P_L^{-\frac{1}{2}}} + \sum_{k=L}^{K} \|\eta_k - h(\mathbf{x}(t_k; \mathbf{u}, \mathbf{p}))\|^2_{V_k^{-\frac{1}{2}}} \right\} \tag{16}$$

$$s.t. \quad \dot{\mathbf{x}}(t) = f(\mathbf{x}(t), \mathbf{u}(t), \mathbf{p}) \tag{17}$$

$$\mathbf{x}_{min} \le \mathbf{x}(t) \le \mathbf{x}_{max} \tag{18}$$

$$\mathbf{p}_{min} \le \mathbf{p} \le \mathbf{p}_{max} \tag{19}$$

$$t \in [t_L, t_K]\,. \tag{20}$$

The second term represents the prediction errors within the horizon and the first term represents the arrival cost (the penalization of a change of the estimates of the initial values of the states and of the parameters), where $(\overline{\mathbf{x}}_L, \overline{\mathbf{p}}_L)$ are the expected values and $P_L \in \mathbb{R}^{(n_x+n_p)\times(n_x+n_p)}$ is the joint covariance matrix of $\mathbf{x}(t_L)$ and $\mathbf{p}_L$. Note that only the initial values of the states and the parameters are free parameters of the optimization problem because no state

noise is assumed within the horizon. The absence of state noise on the horizon is compensated by the simultaneous estimation of key model parameters which is an appropriate assumption since uncertainties are mostly due to model errors and not to disturbances. Furthermore, the inclusion of state noise at each point within the horizon would lead to a large number of degrees of freedom of the estimation and result in a considerably larger optimisation problem with additional $n_x \times (K-L)$ variables that, taking the system dimension into account, would be hard to solve online reliably. From the solution $\mathbf{x}(t_L)$ and $\mathbf{p}$ of the optimization problem, the deterministic model is simulated forward to obtain the current estimated state $\mathbf{x}_K$. The MHE takes upper and lower bounds on the states and on the parameters into account. The expected value and the covariance of $\mathbf{x}(t_L)$ and $\mathbf{p}_L$ in the arrival cost are determined by an Extended Kalman Filter. The smoothed Extended Kalman Filter Robertson et al. (1996) is employed where the recent state estimation $x_{L+1|K}$ and linearizations of the dynamics $G_{L+1|K}$ and the output matrix $C_{L+1|K}$ are utilized. In order to guarantee positive definite matrices in the presence of numerical errors, the smoothed Extended Kalman update is reformulated by two QR-decompositions yielding the equivalent smoothed Extended Kalman Filtering update in square-root formulation Diehl (2002); Kühl et al. (2008).

$$\begin{pmatrix} \overline{\mathbf{x}}_{L+1} \\ \overline{\mathbf{p}}_{L+1} \end{pmatrix} = \begin{pmatrix} \mathbf{x}(t_{L+1}; t_L, \mathbf{x}_{L|K}, \mathbf{u}_L, \mathbf{p}_{L|K}) \\ \mathbf{p}_{L|K} \end{pmatrix} + G_{L|K} R^{-1} Q^T \begin{pmatrix} P_{L|L-1}^{-\frac{1}{2}} \begin{pmatrix} \mathbf{x}_{L|L} - \mathbf{x}_{L|K} \\ \mathbf{p}_{L|L} - \mathbf{p}_{L|K} \end{pmatrix} \\ V_L^{-\frac{1}{2}} (\eta_L - h(\mathbf{x}_{L|K})) \end{pmatrix} \tag{21}$$

$$P_{L+1|L}^{-\frac{1}{2}} = \overline{Q}^T \begin{pmatrix} 0 \\ W^{-\frac{1}{2}} \end{pmatrix} \tag{22}$$

with the linearizations of the dynamics

$$G_{L|K} = \begin{pmatrix} X_{x,L|K} & X_{p,L|K} \\ 0 & I_{n_p} \end{pmatrix} \tag{23}$$

$$X_{x,L|K} = \left. \frac{d\mathbf{x}(t_{L+1}; \mathbf{x}_{L|K}, \mathbf{p}_{L|K})}{d\mathbf{x}} \right|_{L|K} \tag{24}$$

$$X_{p,L|K} = \left. \frac{d\mathbf{x}(t_{L+1}; \mathbf{x}_{L|K}, \mathbf{p}_{L|K})}{d\mathbf{p}} \right|_{L|K}, \tag{25}$$

the linearization of the output

$$H_{L|K} = \begin{pmatrix} H_{x,L|K} & H_{p,L|K} \end{pmatrix} \tag{26}$$

$$H_{x,L|K} = \left. \frac{dh(\mathbf{x}(t_{L+1}; \mathbf{x}_{L|K}, \mathbf{p}_{L|K}))}{d\mathbf{x}} \right|_{L|K} \tag{27}$$

$$H_{p,L|K} = \left. \frac{dh(\mathbf{x}(t_{L+1}; \mathbf{x}_{L|K}, \mathbf{p}_{L|K}))}{d\mathbf{p}} \right|_{L|K}, \tag{28}$$

the QR-decompositions

$$\begin{pmatrix} P_{L|L-1}^{-\frac{1}{2}} \\ V_L^{-\frac{1}{2}} H_{L|K} \end{pmatrix} = (Q|\breve{Q}) \begin{pmatrix} R \\ 0 \end{pmatrix} \tag{29}$$

$$\begin{pmatrix} R \\ -W^{-\frac{1}{2}} G_{L|K} \end{pmatrix} = (\overline{Q}|\widetilde{Q}) \begin{pmatrix} \overline{R} \\ 0 \end{pmatrix}, \tag{30}$$

and the state noise covariance matrix of the states and of the parameters

$$W = \begin{pmatrix} W_x & 0 \\ 0 & W_p \end{pmatrix}. \tag{31}$$

$\mathbf{x}(t_{L+2}; t_{L+1}, \mathbf{x}_{L+1|K}, \mathbf{u}_{L+1}, \mathbf{p}_{L+1|K})$ denotes the prediction of the system based on the recent estimate at $_{L+1|K}$ while $\mathbf{x}_{L+2|L+1}$ is the smoothed prediction.

The MHE has to cope with jumps in the extract and raffinate measurements that are caused by the port switching. In order to obtain a smooth calculation of the gradients with respect to the simulated measurements which exhibit jumps due to the periodic movement of the ports, virtual measurements at constant positions at the outlet of each chromatographic column are included in the mapping h. In order to account for the actual existence of real measurements at the considered point of time k, the corresponding components on the diagonal of the measurement weight V_k^{-1} are set to $\frac{1}{\sigma_v^2}$ while nonexisting measurements cause zero entries on the diagonal of V_k^{-1}. A zero weight can be interpreted as infinite measurement noise. Thus, the correction terms of nonexisting measurements in the smoothed Extended Kalman Filtering update and in the moving horizon are zero. The switching of the measurement weights at the respective extract and raffinate switching times $\delta t_j \tau$ in period m is realized according to the movement of the extract and raffinate port:

$$\overline{V}_{m,j} = M_V \overline{V}_{m,j} \quad j = Ex, Ra \tag{32}$$

$$\overline{V}_m = \overline{V}_{m,Ex} + \overline{V}_{m,Ra} \tag{33}$$

$$V_m = \mathrm{diag}(\overline{V}_{m+1}(1, \ldots, (n_{\mathrm{col}} - 1) * 2), \sigma_v^2, \sigma_v^2). \tag{34}$$

The permutation matrix M_V for shifting the extract and raffinate measurements around the plant for a new period $m+1$ is similar to the permutation matrix M_Q for shifting the port flows. The last two entries of V_k are the variances of the measurements at the internal measurement position (recycle) which are not shifted.

3.2 Multiple-Shooting Real-Time Iteration Scheme for MHE

The moving horizon optimization problem is solved by the multiple shooting method for parameter estimation Bock (1981, 1987). The basic idea of multiple shooting is to subdivide the time horizon into subintervals and to formulate autonomous initial value problems on each individual subinterval which are coupled by continuity conditions. The computational requirements are largely reduced by applying the *real-time iteration* scheme for the multiple shooting method introduced in Diehl et al. (2002, 2004); Diehl (2002) that updates the sensitivity matrices that are necessary to solve the optimization problem before the most recent measurement η_K is available. Another important feature is that the next optimization problem is initialized well at the current solution such that the number of iterations can be reduced to one.

4. RESULTS

For the demonstration of the performance of the moving horizon estimator, the separation of the enantiomer mixture EMD-53986 is considered which is described by a nonlinear adsorption isotherm of extended Langmuir type (12). Enantiomers are chemical molecules that are mirror images of each other, much as one's left and right hands.

The separation of the enantiomers of EMD-53986 was studied experimentally in a joint project by *Merck* (Germany) and Universität Dortmund in 2001. From this work, an accurate simulation model is available. The parameters of the SMB model were taken from Jupke et al. (2002). More details on the process and the model parameters can be found in Jupke (2004). In order to demonstrate the performance of the MHE estimator, a simulation study is presented in which step changes of the Henry coefficients H_A^2, H_B^2 of the nonlinear adsorption isotherm are assumed. The performance of the moving horizon estimator is illustrated by the evolution of the parameters and of the overall state reconstruction error which is defined as

$$J = \sum_{j=1}^{352} (\mathbf{x}(j) - z(j))^2, \tag{35}$$

where z is the true state. The measurements are corrupted by noise with a standard deviation of 0.025 g/l as observed in Jupke et al. (2002). No cross-correlations between the state noises and between the state noises and the parameters were assumed. Since the concentration profiles move around the simulated plant together with the ports, the same noise variances were assumed for each state. The tuning of the moving horizon estimator was performed by varying the covariances of the state variables and of the free parameters. The weighting matrix W incorporates a standard deviation of 0.00433 g/l for the state noise and a parameter standard deviation of 0.0316 for H_{A}^2 and 0.0265 for H_{B}^2: $W^{\frac{1}{2}} = \mathrm{diag}(0.00433, \ldots, 0.00433, 0.0361, 0.0265)$. The initial weight P_0 is set to $0.005 \times W$. The chosen state and parameter noises represent a compromise between the smooth estimation of the states and a quick adaptation of the parameters. The state and parameter bounds are chosen as -0.25 g/l $\leq x \leq 5$ g/l and $0 \leq H_i^2 \leq 50$ to prevent grossly wrong values. The lower bound on the states is chosen such that it remains inactive in the presence of large measurement noise. The sampling time of the estimator is 1/10 of the period length. The moving horizon length is five sampling intervals (half a period).

In the simulation scenario, H_{A}^2 is increased by 10.6% from 19.90 to 22.00 at $t = 14.58$ min while H_{B}^2 is increased by 10.3% from 5.85 to 6.45 at $t = 68.04$ min. It can be seen from figures 4 to 8, that the state is reconstructed correctly in the presence of the parameter variations and that the parameters are also estimated well. The Henry coefficient H_B^2 is estimated faster than H_A^2 due to the stronger excitation of the raffinate dynamics by the parameter perturbation. The axial concentration profiles are reconstructed correctly by the MHE (not show here due to limited space). The MHE is more robust against measurement noise and wrong initializations of the states and parameters than an EKF, see Küpper et al. (2009). The MHE estimator can be applied online, as can be seen from Figure 8. The CPU times are below the sampling rate at all sampling points. The CPU times of the MHE are around 23 s on a standard PC [1], the maximum and minimum values being 28.0 s and 18.5 s. The CPU times for the estimator varies periodically. It was observed that the estimation problem requires a longer computation time when a shift of one of the inlet/outlet ports occurs within the moving horizon.

[1] Intel Xenon CPU 2.8 GHz, 4.0 GB RAM

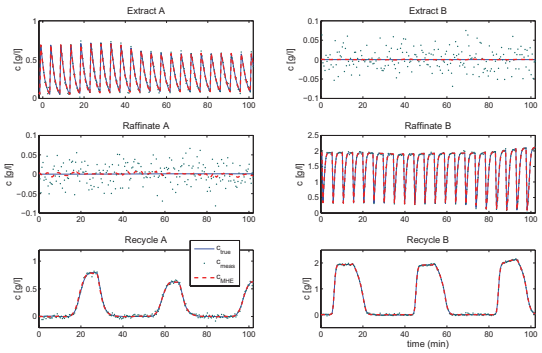


Fig. 4. Measurements (extract, raffinate, recycle)

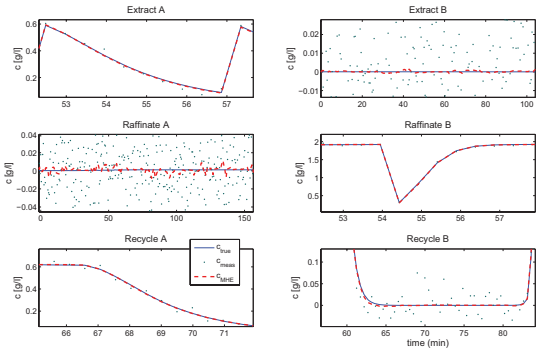


Fig. 5. Enlarged measurements (extract, raffinate, recycle)

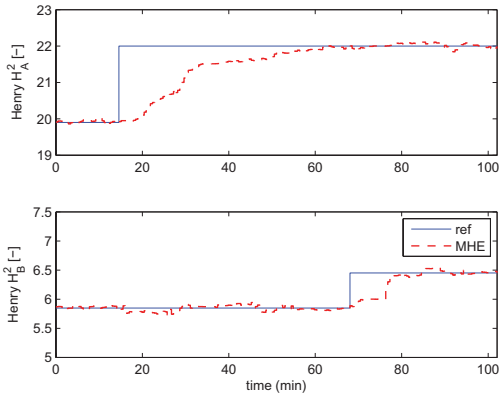


Fig. 6. Parameter estimates

5. ACKNOWLEDGEMENTS

The financial support of the Deutsche Forschungsgemeinschaft (DFG, German Research Concil) in the context of the research cluster "Optimization-based control of chemical processes" (RWTH Aachen, , IWR Heidelberg, Universität Stuttgart, TU Dortmund) is very gratefully acknowledged.

REFERENCES

Alamir, M. and Corriou, J. (2003). Nonlinear receding-horizon state estimation for dispersive adsorption

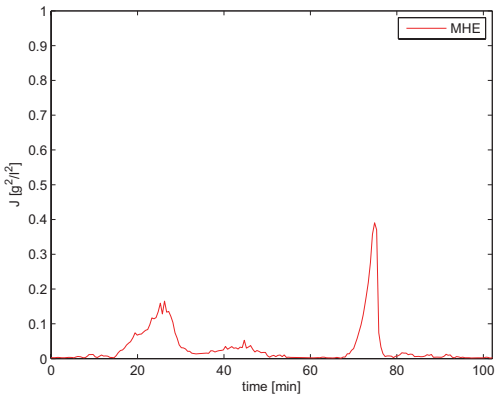


Fig. 7. State reconstruction error

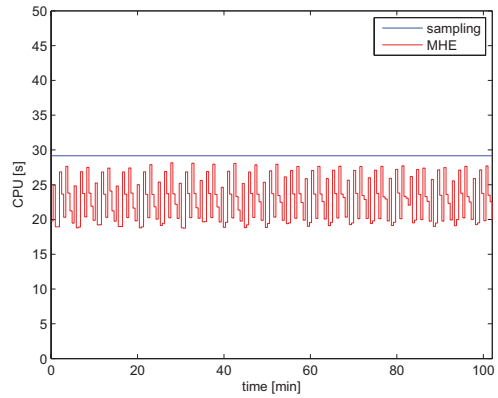


Fig. 8. CPU times of the estimator at each sampling point

columns with nonlinear isotherm. *Journal of Process Control*, 13(6), 517–523.

Bock, H. (1981). Numerical treatment of inverse problems in chemical reaction kinetics. In K. Ebert, P. Deuflhard, and W. Jäger (eds.), *Modelling of Chemical Reaction Systems*, volume 18 of *Springer Series in Chemical Reaction Systems*, 102–125. Springer-Verlag, *Heidelberg*.

Bock, H. (1987). Randwertproblemmethoden zur Parameteridentifizierung in Systemen nichtlinearer Differentialgleichungen. volume 183 of *Bonner Mathematische Schriften*. Universität Bonn, *Bonn*.

Diehl, M. (2002). Real-time optimization of large scale nonlinear processes. volume 920 of *Fortschritt-Berichte VDI Reihe 8, Mess-, Steuerungs- und Regelungstechnik*. VDI Verlag, *Düsseldorf*.

Diehl, M., Bock, H., Schlöder, J., Findeisen, R., Nagy, Z., and Allgöwer, F. (2002). Real-time optimization and nonlinear model predictive control of processes governed by differential-algebraic equations. *Journal of Process Control*, 12(4), 577–585.

Diehl, M., Bock, H., and Schlöder, J. (2004). A real-time iteration scheme for nonlinear optimization in optimal feedback control. *SIAM Journal on Control and Optimization*, 43(5), 1714–1736.

Diehl, M., Kühl, P., Bock, H., and Schlöder, J. (2006). Schnelle Algorithmen für die Zustands- und Parameterschätzung auf bewegten Horizonten. *at-Automatierungstechnik*, 54(12), 602–613.

Dünnebier, G. and Klatt, K.U. (2000). Modelling and simulation of nonlinear chromatographic separation processes: A comparison of different modelling aspects. *Chemical Engineering Science*, 55(2), 373–380.

Gu, T. (1995). Mathematical Modelling and Scale-up of Liquid Chromatography. *Springer Verlag, New York*.

Jupke, A. (2004). Experimentelle Modellvalidierung und modellbasierte Auslegung von Simulated Moving Bed (SMB) Chromatographieverfahren, Dr.-Ing. Dissertation, Fachbereich Bio- und Chemieingenieurwesen, Universität Dortmund. *VDI Reihe 3, Nr. 807*.

Jupke, A., Epping, A., and Schmidt-Traub, H. (2002). Optimal design of batch and simulated moving bed chromatographic separation processes. *Journal of Chromatography A*, 944(1-2), 93–117.

Kühl, P., Diehl, M., Kraus, T., Bock, H., and Schlöder, J. (2008). A real-time algorithm for moving horizon state and parameter estimation. *Journal of Process Control*. (submitted).

Kleinert, T. and Lunze, J. (2005). Modelling and state observation of Simulated Moving Bed processes based on explicit functional wave form description. *Mathematics and Computers in Simulation*, 68(3), 235–270.

Kloppenburg, E. and Gilles, E. (1999). Automatic control of the simulated moving bed process for C8 aromatics separation using asymptotically exact input/output linearization. *Journal of Process Control*, 9, 41–50.

Küpper, A., Diehl, M., Schlöder, J., Bock, H., and Engell, S. (2009). Efficient moving horizon state and parameter estimation for smb processes. *Journal of Process Control*, doi:10.1016/j.jprocont.2008.10.004.

Küpper, A. and Engell, S. (2006). Parameter and state estimation in chromatographic SMB processes with individual columns and nonlinear adsorption isotherms. *Proc. of the IFAC International Symposium of Advanced Control of Chemical Processes, Gramado*, 611–616.

Mangold, M., Lauschke, G., Schaffner, J., Zeitz, M., and Gilles, E.D. (1994). State and parameter estimation for adsorption columns by nonlinear distributed parameter state observers. *Journal of Process Control*, 4(3), 163–172.

Robertson, D., Lee, J., and Rawlings, J. (1996). A moving horizon-based approach for least-squares estimation. *AIChE Journal*, 42(8), 2209–2224.

Toumi, A., Engell, S., Ludemann-Hombourger, O., Nicoud, R.M., and Bailly, M. (2003). Optimization of simulated moving bed and Varicol processes. *Journal of Chromatography A*, 1-2, 15–31.

Toumi, A., Hanisch, F., and Engell, S. (2002). Optimal operation of continuous chromatographic processes: Mathematical optimization of the VARICOL process. *Ind. Eng. Chem. Res.*, 41(17), 4328–4337.

State estimation for large-scale wastewater treatment plants ⋆

Jan Busch [*] [1] **Peter Kühl** [**] [2] **Johannes P. Schlöder** [**]
Hans Georg Bock [**] **Wolfgang Marquardt** [*]

[*] *AVT Process Systems Engineering, RWTH Aachen University, Germany*
[**] *IWR, Heidelberg University, Germany*

AbstractMany relevant process states in wastewater treatment are not measurable, or their measurements are subject to considerable uncertainty. This poses a serious problem for process monitoring and control. Model-based state estimation can provide estimates of the unknown states and increase the reliability of measurements. In this paper, an integrated approach is presented for the estimation problem employing unconventional, but technically feasible sensor networks. Using the ASM1 model in the reference scenario BSM1, the estimators EKF and MHE are evaluated. Very good estimation results for the system comprising of 78 states are found.

Keywords: state estimation, MHE, EKF, wastewater treatment, ASM, BSM1

1. INTRODUCTION

One of the key challenges in the operation of activated sludge wastewater treatment plants (WWTP) is the uncertainty about relevant process state values. E. g. the concentrations of active biomass and of soluble substrate are not measurable online, but they considerably influence process behavior. Some states such as the concentration of total suspended solids are measurable, but their measurements involve significant measurement errors. Reliable estimates of these states are of great value for different operational tasks such as process monitoring, online simulation, and advanced multi-variable control. They are a necessity for model-based control approaches based on dynamic process models (e. g. Busch et al., 2007). Model-based state estimation is one alternative to obtain such estimates. For a given process model, its success depends on the choice of a suitable hardware sensor network and of an appropriate estimation method.

The intention of this paper is to present sophisticated solutions to the state estimation problem for large-scale WWTP and to investigate two distinct state estimation approaches from the practitioner's point of view. First, an optimization-based approach determines the cheapest hardware sensor network that is required for the state estimation task. Second, Extended Kalman Filtering (EKF) and Moving Horizon Estimation (MHE) are employed to estimate the unknown model states of the large-scale WWTP model ASM1 employed in the BSM1 reference scenario (Copp, 2002). Large measurement errors, plant/model-mismatch, and unknown inflow concentrations are considered.

⋆ We thank the German research foundation (DFG) for the financial support in the project "Optimization-based process control of chemical processes" (grant MA 1188/27-1). Also BMBF grant 03BONCHD is gratefully acknowledged.

[1] present address: Bayer Technology Services, Leverkusen, Germany
[2] present address: BASF SE, Ludwigshafen, Germany

State estimation aims at statistically optimal estimates of measurable and unmeasurable process states. Dochain (2003) provides an overview of state and parameter estimation for chemical and biochemical processes focusing on small models. Lubenova et al. (2003) use an adaptive observer for a bioprocess models with 5 states. Goffaux and Vande Wouwer (2005) compare an asymptotic observer, an EKF, and a particle filter (PF) for a bioprocess model with 4 states. A model with approximately 40 states based on the ASM1 successor ASM3 is considered by Chai et al. (2007), who evaluate a KF, an EKF, and an unscented KF (UKF). No rigorous MHE implementation for WWTP has been reported.

Generally speaking, observers prove to be efficient for small-scale models with maybe up to 10 states. For larger models, observer design becomes challenging. An exception are asymptotic observers, which exhibit slow convergence of the estimates to the true values, but which do not require kinetic models. The EKF is the standard choice for large-scale models. It is easy to implement and much experience is available concerning its design and tuning. It is not clear whether the related UKF and PF can significantly outperform the EKF in practical implementations. The MHE is a promising option, but it is not clear whether its increased implementation effort is justified by better estimation results in WWTP applications. Large-scale simulation case studies are rare, and real-life case studies are not available. Also, while the properties of the hardware sensor network are decisive for the success of any state estimation approach, this aspect has not been treated much with respect to WWTP applications.

Ideally, the choice of a sensor network, of a process model, and of the estimator should be considered as an integrated problem. This is beyond our possibilities today. The sensor networks used in this study are obtained by a simple optimization-based sensor network design approach. An observable system for a given large-scale plant model

involving 78 differential states is obtained. An EKF and an MHE are then employed as state estimators.

2. PROCESS AND PROCESS MODEL

The simulation study is based on the BSM1 (Copp, 2002), which has been developed as a benchmark scenario for the evaluation and comparison of different control approaches for WWTP. The plant layout is depicted in Fig. 1. Q_i and $\mathbf{Z}_i$ refer to the flow rate and vector of concentrations for stream i. The inflow is mixed with two recycle streams before entering the plant. Two denitrification basins (each $1000\,\mathrm{m}^3$) are followed by three aerated nitrification basins (each $1333\,\mathrm{m}^3$). The first recycle a is withdrawn from the last nitrification basin. The settler used in the BSM1 is replaced by a membrane filtration unit, which is located in a separate $250\,\mathrm{m}^3$ basin and which is modeled as an ideal splitter. The product stream e as well as a second recycle r and a waste stream w leave the membrane basin. All basins are assumed to be well-mixed.

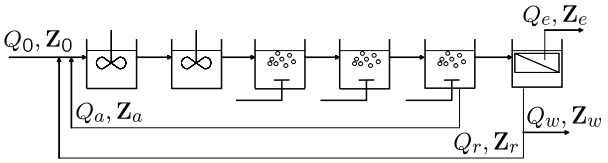


Figure 1. Modified BSM1 plant layout.

The degradation processes in the five biological basins are described by the ASM1 (Henze et al., 1987) with parameters taken from Copp (2002). The ASM1 describes 8 reactions and the component concentrations of inert soluble matter S_I, soluble substrate S_S, inert particulate matter X_I, particulate substrate X_S, heterotrophic biomass $X_{B,H}$, autotrophic biomass $X_{B,A}$, particulate inert metabolism products X_P, dissolved oxygen (DO) S_O, nitrate S_{NO}, ammonia S_{NH}, soluble organic nitrogen S_{ND}, particulate organic nitrogen X_{ND}, and the alkalinity S_{ALK}. The resulting model comprising mass balances and the kinetic model contains 78 differential states. It is formulated as a semi-explicit differential-algebraic model according to

$$\dot{\mathbf{x}} = \mathbf{f}\left(\mathbf{x}, \mathbf{z}, \mathbf{u}, \mathbf{p}\right) \,, \tag{1}$$

$$0 = \mathbf{g}\left(\mathbf{x}, \mathbf{z}, \mathbf{u}, \mathbf{p}\right) \,, \tag{2}$$

$$\mathbf{y} = \mathbf{M} \cdot \mathbf{x} \,. \tag{3}$$

$\mathbf{x}$ are differential and $\mathbf{z}$ are algebraic states, $\mathbf{u}$ are the manipulated variables, and $\mathbf{p}$ are the parameters. $\mathbf{y}$ are the measurable outputs and $\mathbf{M}$ is the measurement matrix. Note that for the BSM1 scenario, $\mathbf{g}(\cdot)$ represents defining equations that can explicitly be solved for $\mathbf{z}$. Generally $\mathbf{g}(\cdot)$ suffices to be of index 1.

The BSM1 benchmark describes a dry weather scenario for a period of 100 days with constant manipulated variables, inflow rates, and inflow concentrations to reach a steady state. This is followed by a period of 14 days with dynamic inflow conditions. One of the three different dynamic scenarios, the *storm scenario*, is used in this paper. It is characterized by dry weather inflow superposed by storm events on days 9 and 11. Exemplarily, the corresponding inflow rate and ammonia inflow concentration are depicted in Fig. 2.

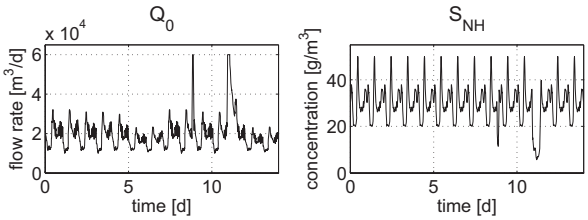


Figure 2. Inflow rate Q_0 and ammonia inflow concentration $Z_{0,S_{NH}}$.

3. SENSOR NETWORK DESIGN

The success of a state estimation approach depends on the process model, on the state estimation algorithm, and on the sensor network which supplies the measurements. Optimal sensor network design aims at the sensor network which leads to optimal state estimates at limited cost, or similarly, reliable state estimates at minimum cost (Singh and Hahn, 2005). So far, systematic approaches to this important aspect of state estimation have been neglected in the literature on WWTP applications. Rather, the sensor network is chosen based on experience and intuition.

The approach to obtain the sensor networks as used in this study is outlined in the following, details are presented in Busch et al. (2009). A sensor network is fully defined by the measurement matrix $\mathbf{M}$ (Eq. (3)), which relates process states $\mathbf{x}$ to measurements $\mathbf{y}$. By assigning prices to the measurement hardware, a cost function $\phi = \phi\left(\mathbf{M}\right)$ is obtained, which describes the cost of the sensor network. The relevant constraint for the sensor network is that it needs to yield an observable system. Hence, the nonlinear process model is linearized at many instances along a typical process trajectory, and observability is checked for each of these instances by a suitable criterium. Finally, a genetic optimization algorithm is employed to find the sensor network with the minimum cost $\phi\left(\mathbf{M}\right)$ which still fulfills the observability constraints (Heyen and Gerkens, 2002). The approach is applied to the simulation scenario described in Section 2. Considering 8 technically feasible measurements in 6 basins gives a total of $2^{6\cdot 8} \approx 2.8 \cdot 10^{14}$ measurement configurations.

The following sensor network is found to give observability at minimum cost: $COD_1, S_{ALK,1}, S_{O,2}, X_{TS,5}$, where the numeric index refers to the basin number. COD is the chemical oxygen demand. This result is quite surprising, as it implies that only four hardware sensors suffice to estimate all 78 model states. Some standard hardware sensors, which are commonly available at WWTP, are added to the sensor network. These are DO sensors in the aerated basins as well as nitrate, ammonia, alkalinity, and COD measurements in the effluent.

4. STATE ESTIMATORS

State estimation refers to retrieving all states of a dynamic system in real-time by utilizing available measurements, possibly in combination with a process model. While the state estimation problem is largely solved for *linear systems*, e. g. , by the Kalman Filter, the problem becomes significantly more difficult for non-linear systems. Most methods are extensions of linear state estimators, such

as the extended Kalman Filter (EKF), described e. g. in Becerra et al. (2001). A *non-linear* version of MHE is presented in Rao et al. (2003). A comparison of EKF and non-linear MHE applied to the BSM1 scenario is presented in Section 5. In the following, main principles and implementation details are reviewed.

4.1 Extended Kalman Filter

The Kalman Filter is a recursive method for state estimation. It consists of a prediction step (time update) and a measurement update. Past data is summarized and carried on by means of suitable statistics. For a non-linear system in discrete-time with measurement noise $\boldsymbol{\upsilon}_k \sim \mathcal{N}(0, \tilde{\mathbf{V}})$ and process noise $\boldsymbol{\mu}_k \sim \mathcal{N}(0, \tilde{\mathbf{W}})$

$$\mathbf{x}_k = \mathbf{f}_k\left(\mathbf{x}_{k-1}, \mathbf{z}_{k-1}, \mathbf{u}_{k-1}, \mathbf{p}_{k-1}\right) + \boldsymbol{\mu}_k\,, \tag{4}$$

$$\mathbf{0} = \mathbf{g}_k\left(\mathbf{x}_{k-1}, \mathbf{z}_{k-1}, \mathbf{u}_{k-1}, \mathbf{p}_{k-1}\right), \tag{5}$$

$$\mathbf{y}_k = \mathbf{M}\cdot\mathbf{x}_k + \boldsymbol{\upsilon}_k\,, \tag{6}$$

where k denotes the sampling instant, the respective filter equations in their most common form are:

Time update:

$$\hat{\mathbf{x}}_k^- = \mathbf{f}_k\left(\hat{\mathbf{x}}_{k-1}, \mathbf{z}_{k-1}, \mathbf{u}_{k-1}, \mathbf{p}_{k-1}\right), \tag{7a}$$

$$\mathbf{P}_k^- = \left.\frac{\partial \mathbf{f}_k}{\partial \mathbf{x}_{k-1}}\right|_{\hat{\mathbf{x}}_{k-1}} \cdot \mathbf{P}_{k-1} \cdot \left.\frac{\partial \mathbf{f}_k}{\partial \mathbf{x}_{k-1}}\right|_{\hat{\mathbf{x}}_{k-1}}^{\mathrm{T}} + \mathbf{W}\,, \tag{7b}$$

Measurement update:

$$\mathbf{K}_k = \mathbf{P}_k^- \cdot \mathbf{M}_d^{\mathrm{T}} \cdot \left(\mathbf{M}_d \cdot \mathbf{P}_k^- \cdot \mathbf{M}_d^{\mathrm{T}} + \mathbf{V}\right)^{-1}, \tag{8a}$$

$$\mathbf{P}_k = \left(\mathbb{I} - \mathbf{K}_k \cdot \mathbf{M}_d\right) \cdot \mathbf{P}_k^-\,, \tag{8b}$$

$$\hat{\mathbf{x}}_k = \hat{\mathbf{x}}_k^- + \mathbf{K}_k \cdot \left(\mathbf{y}_k - \mathbf{M}_d \cdot \hat{\mathbf{x}}_k^-\right), \tag{8c}$$

$$\mathbf{0} = \mathbf{g}_k\left(\hat{\mathbf{x}}_k, \hat{\mathbf{z}}_k, \mathbf{u}_k, \mathbf{p}_k\right). \tag{8d}$$

$\mathbf{f}_k$ typically represents a numerical integration of the continuous system Eq. (1) from time t_{k-1} to t_k with initial values $\mathbf{x}_{k-1}$. The matrix $\mathbf{P}_k$ is the covariance matrix associated with the state estimates $\hat{\mathbf{x}}_k$ at sampling time k. It reflects the confidence one can have in this estimate. The matrices $\mathbf{V}$ and $\mathbf{W}$ describe the assumed covariances of measurement noise and process noise, respectively. The Kalman Filter gain $\mathbf{K}_k$ then reflects the trade-off between the measurements and the process model.

4.2 Moving Horizon Estimation

A drawback of most estimation methods is that they cannot deal with known constraints on the estimated states. In the MHE scheme, such constraints are naturally incorporated in the optimization problem. The formulation also allows to additionally estimate process parameters without reformulating them as dummy states.

Unlike the EKF, the MHE uses more than just the most recent measurements: At a certain time t_j a number of $M+1$ measurements $(\mathbf{y}_{j-M}, \ldots, \mathbf{y}_j)$ associated with past time instants $t_{j-M} < \ldots < t_j$ are explicitly used for estimation. The length L of the time horizon $[t_j, \ldots, t_{j-M}]$ is defined as $L := j - M$. It is assumed that measurement and process noise are normally distributed with zero mean and covariance matrices $\mathbf{V}$ and $\mathbf{W}$. Additionally, a Gaussian distribution is assumed for $\mathbf{x}(t_L)$ and $\mathbf{p}$ at the beginning of the horizon, with expectation value $(\bar{\mathbf{x}}_L, \bar{\mathbf{p}}_L)$ and a block-diagonal covariance matrix $\boldsymbol{\Pi}_L$ with block elements $\boldsymbol{\Pi}_{\bar{x},L}$ and $\boldsymbol{\Pi}_{\bar{p},L}$.

The state estimation problem to be solved at time t_k – given the measurements $\mathbf{y}_j$ for $j = L, L+1, \ldots, k$, the known input $\mathbf{u}(t)$ for $t \in [t_L, t_k]$ and given $(\bar{\mathbf{x}}_L, \bar{\mathbf{p}}_L)$ and $\mathbf{P}_L$ – has the following form:

$$\min_{\mathbf{x}(\cdot),\mathbf{p}} \Big(\|\mathbf{x}(t_L) - \bar{\mathbf{x}}_L\|^2_{\boldsymbol{\Pi}_{\bar{x},L}} + \|\mathbf{p} - \bar{\mathbf{p}}_L\|^2_{\boldsymbol{\Pi}_{\bar{p},L}} + \sum_{j=L}^{k} \|\mathbf{y}_j - \mathbf{M}\cdot\mathbf{x}(t_j)\|^2_{\mathbf{V}} \Big) \tag{9}$$

$$\text{s.t.} \quad \dot{\mathbf{x}}(t) = \mathbf{f}(\mathbf{x}(t), \mathbf{z}(t), \mathbf{u}(t), \mathbf{p}), \quad t \in [t_L, t_k], \tag{10a}$$

$$\mathbf{0} = \mathbf{g}(\mathbf{x}(t), \mathbf{z}(t), \mathbf{u}(t), \mathbf{p}), \tag{10b}$$

$$\mathbf{x}_{\min} \le \mathbf{x}(t) \le \mathbf{x}_{\max}, \tag{10c}$$

$$\mathbf{p}_{\min} \le \mathbf{p} \le \mathbf{p}_{\max}, \tag{10d}$$

where the applied norm is defined as $\|\mathbf{x}\|^2_{\mathbf{V}} := \mathbf{x}^{\mathrm{T}}\mathbf{V}^{-1}\mathbf{x}$.

At each new sampling time t_k, one new measurement vector $\mathbf{y}_k$ enters the set of measurements, while the last one $\mathbf{y}_L$ becomes $\mathbf{y}_{L-1}$ and drops out of the horizon.

The initial weight terms $\|\mathbf{x}(t_L) - \bar{\mathbf{x}}_L\|^2_{\boldsymbol{\Pi}_{\bar{x},L}}$ and $\|\mathbf{p} - \bar{\mathbf{p}}_L\|^2_{\boldsymbol{\Pi}_{\bar{p},L}}$ (often called "arrival costs") summarize information in the MHE problem prior to the horizon beginning at time t_L and also reflect a cumulated effect of process noise on the process. A typical approach is to compute the arrival costs by Kalman Filter updates of $\bar{\mathbf{x}}_L$ and $\bar{\mathbf{p}}_L$. As for the optimal length of the estimation horizon, no general results are available, yet. Horizon length and the weighting matrices $\mathbf{V}^{-1}$, $\boldsymbol{\Pi}^{-1}$ are the tuning parameters. Note that an extended MHE formulation exists that explicitly incorporates process noise (Rao et al., 2003; Diehl et al., 2006).

A necessity for the MHE scheme to work is a fast and reliable numerical scheme for the constrained non-linear dynamic optimization problem (9). The implementation in this work makes use of MUSCOD-II (Leineweber et al., 2003), which is based on a direct multiple shooting approach, see, e. g. , (Bock et al., 2007). For real-time feasibility, the least-squares problem at each time instant t_k is not solved to convergence. Instead, only one Gauss-Newton iteration is performed, combined with a meaningful shift of the problem variables. More information on this so-called *real-time iteration approach* along with other implementational details can be found in Diehl et al. (2006).

5. CASE STUDY

EKF and MHE are applied to the process model and scenario described in Section 2 and the sensor network calculated in Section 3. The estimation task was made increasingly difficult to evaluate the estimation performance under nominal and more realistic conditions.

Only little effort has been devoted to the fine-tuning of the estimators. This is intentional, since the aim of the study is to investigate the practical applicability and general performance of the two estimation methods. The tuning matrices $\mathbf{W}$ and $\mathbf{V}$ for the EKF and the MHE reflect covariances based on an assumed standard deviation of 5% of the initial values $\mathbf{x}_0$ and "initial measurements" $\mathbf{M}\cdot\mathbf{x}_0$. The MHE uses an estimation horizon of 5 measurement samples.

The initial guess for $\hat{\mathbf{x}}_0$ is deliberately set to $1.3 \cdot \mathbf{x}_0$ to introduce a strong initial offset. The measurements are corrupted by white noise $\boldsymbol{\upsilon}$ with a standard deviation of 5% of the initial measurements: $\mathbf{y}_k = \mathbf{M} \cdot \mathbf{x}_k + \boldsymbol{\upsilon}_k$. The sampling interval is set to 15 minutes. In the following, the quality of the estimation results will be illustrated by the estimates of the third basin, which is the one with the least measurements (only DO concentration). The DO concentration is not visualized as the estimates always closely follow the true values.

5.1 Nominal process

In the first scenario, no process noise is added and perfect knowledge of the inflow rate and concentrations is assumed. The estimated state values quickly converge from their initial offset to the true values (not shown). Only the concentration of X_P shows some occasional offset. The root of the cumulated squared relative error (RCSE) averaged over all J samplings is used in the following as a measure to compare the overall estimation performance:

$$\mathrm{RCSE} = \frac{1}{J} \sum_{k=1}^{J} \sqrt{\sum_{i=1}^{N} \left(\frac{\hat{x}_{i,k} - x_{i,k}}{x_{i,k}} \right)^2}, \qquad (11)$$

where J is the number of samples $x_{i,k}$ and $\hat{x}_{i,k}$ and N is the number of states. The RCSE values for the different simulation case studies are stated in Table 1. For the EKF and the MHE with known inputs and no process noise, RCSE of 0.3 and 0.4 are obtained, respectively.

Table 1. RCSE for the estimated states of different simulation scenarios and estimators.

Estimator	Known inputs, no process noise	Known inputs, process noise	Unknown inputs, process noise
EKF	0.3	0.7	1.8
MHE	0.4	0.8	1.6

5.2 Process noise

Process noise is added to introduce plant/model-mismatch to the problem. The process noise $\boldsymbol{\mu}$ has zero mean and a standard deviation of 5% of the initial states and enters the discrete time simulation model according to

$$\mathbf{x}_{k+1} = \mathbf{f}\left(\mathbf{x}_k, \mathbf{z}_k, \mathbf{u}_k, \mathbf{p}_k\right) + \boldsymbol{\mu}_k \,, \qquad (12a)$$

$$\mathbf{0} = \mathbf{g}\left(\mathbf{x}_k, \mathbf{z}_k, \mathbf{u}_k, \mathbf{p}_k\right). \qquad (12b)$$

Noise-induced negative states representing concentrations are set to zero to ensure that the equations remain physically feasible. The estimation results are similar for the EKF and the MHE. Exemplarily, Fig. 3 shows the results for the third basin using the MHE. Deviations from the true trajectories are observed, but the estimation result averaged over all samples remains satisfactory for both estimators. The RCSE of the EKF and the MHE changes from 0.3 to 0.7 and from 0.4 to 0.8, respectively (Table 1). This result is not surprising, as the process noise now deviates the measured outputs, complicated even further by the process nonlinearities which are not fully captured by the estimators.

5.3 Unknown inflow concentrations

Up to now it has been assumed that the inflow rate and concentrations are perfectly known. This assumption is not realistic. While the inflow rate is indeed well-known, at least part of the inflow concentrations are not. Typically historic data is employed to obtain daily, weekly, and yearly trends and patterns of e. g. the concentrations or the composition of the COD. However, a substantial bias between these predictions and the real inflow concentrations must be expected. A worst case situation is considered in the following: The inflow concentrations of soluble inert matter S_I, soluble substrate S_S, particulate inert matter X_I, particulate substrate X_S, heterotrophic biomass $X_{B,H}$, as well as soluble S_{ND} and particulate organic nitrogen X_{ND} are treated as unknown model inputs. The inflow concentrations of DO S_O, autotrophic biomass $X_{B,A}$, metabolism products X_P, and nitrate X_{NH} are set to zero, and the alkalinity S_{ALK} is set to 7, which corresponds to typical inflow characteristics as well as to the BSM1. The inflow rate Q_0 and the inflow ammonia concentration S_{NH} are measurable. The unknown inputs need to be estimated together with the unknown states.

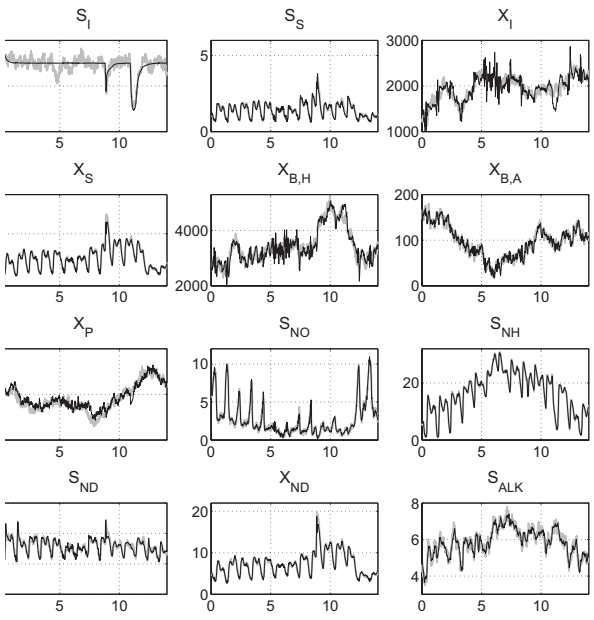


Figure 3. MHE with 5% process noise, third basin. The x-axis shows the time in days, and the y-axis shows the concentrations in [g/m^3]. The alkalinity S_{ALK} is dimensionless. The graphs show the true process state values (light grey) and the estimates (black).

First, a new sensor network is determined by applying the optimization procedure outlined in Section 3 to an extended model, which considers the unknown inflow concentrations as additional model states (Busch et al., 2009). The resulting sensor network is more complex than the network used for the estimation of the nominal process, but still technically and economically feasible:

$$X_{TS,1}, S_{ALK,1}, BOD_2, BOD_3, S_{O,3}, S_{ALK,3}\,, \\ S_{ALK,4}, COD_5, COD_6\,,$$

where BOD is the biological oxygen demand. The same standard measurements as discussed in Section 3 are added to the sensor network.

The estimation of the model parameters such as input concentrations is an integrated part of the MHE implementation and thus can be pursued very easily (see Section 4). For the EKF, the effort is slightly higher. Here, to additionally estimate process parameters, these have to be formulated as additional differential states $\mathbf{x}_p$ obeying the trivial differential equation $\dot{\mathbf{x}}_p = \mathbf{0}$ with initial values $\mathbf{x}_p(0) = \mathbf{p}$. The EKF then estimates the augmented state vector $(\mathbf{x}^{\mathrm{T}}\,\mathbf{x}_p^{\mathrm{T}})^{\mathrm{T}}$. Note that the covariance matrix $\mathbf{W}$ has to be adapted to the new state vector. The expected process noise standard deviation of the unknown parameters is specified as 5% of their nominal values. The initial guess for the inflow concentrations is also disturbed by +30%. Fig. 4 depicts the estimated states for the third basin. The estimation performance is again satisfactorily except for two states. The estimation of inert particulate matter X_I shows considerable offset from the true values. This is, however, not severe, as inert matter does not affect the reaction kinetics and is hence irrelevant for process prediction. The second state to exhibit a significant offset is the concentration of heterotrophic biomass $X_{B,H}$. This is more serious as heterotrophic biomass is responsible for the degradation of substrate and nitrate. Whether the offset is critical, e. g. in model-based control approaches, needs to be evaluated in future research. Fine-tuning of the estimator might further minimize the deviation. The overall RCSE is 1.8 for the states (Table 1) and 2.0 for the parameters.

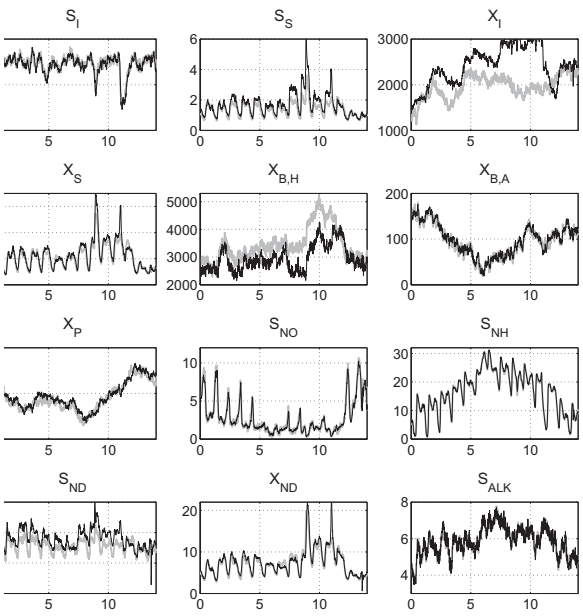


Figure 4. EKF with unknown inflow concentrations and 5% process noise, third basin. The x-axis shows the time in days, and the y-axis shows the concentrations in [g/m^3]. The alkalinity S_{ALK} is dimensionless. The graphs show the true process state values (light grey) and the estimates (black).

Fig. 5 shows the estimation results for the states in the third basin as obtained by the MHE. The results do not differ much from those of the EKF. Again, the two states inert particulate matter X_I and heterotrophic biomass $X_{B,H}$ show the largest deviations. From visual inspection, the first seems to stay closer to the true value but then exhibits a sudden and sharp drop which is not present in the real trend. The overall RCSE for this case is 1.6 (Table 1) and hence slightly better than for the EKF. The estimated parameters achieve an RCSE of 2.1.

The estimated inflow concentrations are depicted exemplarily for the EKF in Fig. 6. All estimates exhibit high-frequent oscillations, which could probably also be improved by fine-tuning of the estimators. The estimation of the concentration of heterotrophic biomass $X_{B,H}$ again shows a stronger offset during days 11 to 13 following the second storm event but returns to the true value eventually. The graphs of inert particulate matter X_I and particulate organic nitrogen X_{ND} show that the estimates are not able to follow sudden concentration peaks (day 9).

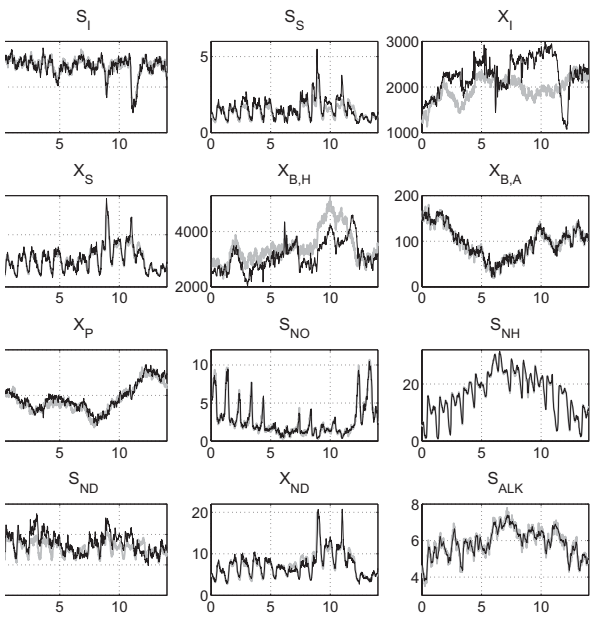


Figure 5. MHE with unknown inflow concentrations and 5% process noise, third basin. The x-axis shows the time in days, and the y-axis shows the concentrations in [g/m^3]. The alkalinity S_{ALK} is dimensionless. The graphs show the true process state values (light grey) and the estimates (black).

The main trends in the inflow data are captured well, but it is not clear especially with respect to the concentrations of inert particulate matter X_I and heterotrophic biomass $X_{B,H}$ whether these parameter are actually observable. To clarify the issue, additional scenarios have been calculated which show that the parameters are indeed observable, but that their influence on the noisy process and process measurements is small, so that it is not possible to resolve higher frequent variations Busch et al. (2009).

5.4 Computation times

A general belief that can often be found is that optimization-based estimation methods such as MHE are impractical

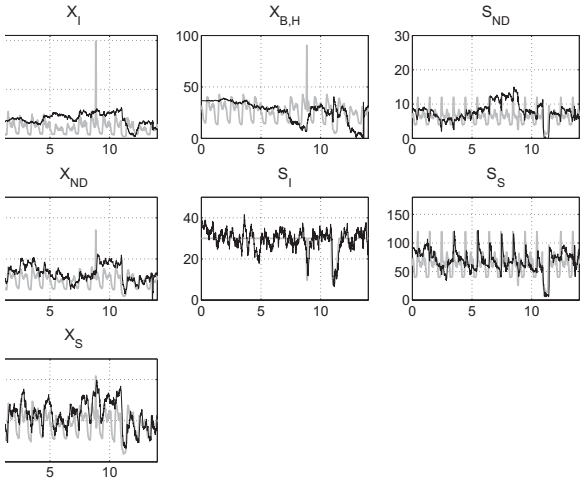


Figure 6. EKF with unknown inflow concentrations and 5% process noise. The x-axis shows the time in days, and the y-axis shows the inflow concentrations in [g/m^3]. The graphs show the true process state values (light grey) and the estimates (black).

because of the excessive computation times to be expected. Indeed the time required to solve a constrained optimization problem to full convergence will necessarily be larger than recursively solving the corresponding unconstrained problem. However, the numerical approach sketched in Section 4 can significantly reduce the computation times. In the case studies described, the average computation time for the MHE was in the range of a few seconds with a maximum lower than 10 seconds on a Pentium 4 machine with 2.8 GHz, 1024 kB L2 cache, 1 GB RAM under Suse Linux 9.3. This is by far fast enough for the estimation tasks for WWTP.

6. CONCLUSIONS

In this paper inflow and state estimation approaches for large-scale wastewater treatment plants are presented. The process is based on the reference scenario BSM1 and employs the dynamic, non-linear process model ASM1. The two prominent state estimators EKF and MHE are evaluated. Large process and measurement disturbances as well as unknown influent conditions have been considered.

The results show that it is possible to yield a fully observable system with an unconventional sensor network of moderate complexity. Both the EKF and the MHE show good estimation performance even in difficult conditions. The EKF shows a marginally better performance for the scenarios with known inflow concentrations. For unknown inflow concentrations, the MHE delivers slightly better state estimates. These do not fully justify the higher implementational effort for the MHE. However, its simple and straightforward handling of unknown inflow conditions and parameters is an advantage over the EKF. The computation times presented here show that the EKF as well as the MHE are real-time feasible for WWTP.

REFERENCES

Becerra, V.M., Roberts, P.D., and Griffiths, G.W. (2001). Applying the extended Kalman Filter to systems described by nonlinear differential-algebraic equations. *Control Eng. Pract.*, 9, 267–281.

Bock, H., Kostina, E., and Schlöder, J.P. (2007). Numerical methods for parameter estimation in nonlinear daes. *GAMM Mitteilungen*, 30/2, 352–375.

Busch, J., Kuehl, P., Gerkens, C., Marquardt, W., Schlöder, J.P., and Bock, H.G. (2009). State estimation for large-scale wastewater treatment plants. *Wat. Res.* (in preparation).

Busch, J., Oldenburg, J., Santos, M., Cruse, A., and Marquardt, W. (2007). Dynamic predictive scheduling of operational strategies for continuous processes using mixed-logic dynamic optimization. *Comput. Chem. Eng.*, 31(5–6), 574–587.

Chai, Q., Furenes, B., and Lie, B. (2007). Comparison of state estimation techniques, applied to a biological wastewater treatment process. In *Proceedings of the 10th IFAC Symposium on Computer Applications in Biotechnology*, 353–358. Cancun, Mexico.

Copp, J.B. (ed.) (2002). *The COST Simulation Benchmark. Description and Simulator Manual*. Office for Official Publications of the European Communities, Luxembourg.

Diehl, M., Kuehl, P., Bock, H.G., and Schlöder, J.P. (2006). Schnelle Algorithmen für die Zustands- und Parameterschätzung auf bewegten Horizonten. *Automatisierungstechnik*, 54(12), 602–613.

Dochain, D. (2003). State and parameter estimation in chemical and biochemical processes: a tutorial. *J. Process Contr.*, 13, 801–818.

Goffaux, G. and Vande Wouwer, A. (2005). Bioprocess state estimation: Some classical and less classical approaches. In T.M. et al. (ed.), *Control and Observer Design*, 111–128. Springer, Berlin, Heidelberg.

Henze, M., Grady Jr., C.P.L., Gujer, W., Marais, G.V.R., and Matsuo, T. (1987). A general model for single-sludge wastewater treatment systems. *Water Res.*, 21(5), 505–515.

Heyen, G. and Gerkens, C. (2002). Application d'algorithmes génétiques à la synthèse de systémes de mesure redondants. In *Proceedings of SIMO 2002 Congress*. Toulouse, France.

Leineweber, D.B., Schäfer, A., Bock, H.G., and Schlöder, J.P. (2003). An efficient multiple shooting based reduced SQP strategy for large-scale dynamic process optimization. Part II: Software aspects and applications. *Comput. Chem. Eng.*, 27, 167–174.

Lubenova, V., Rocha, I., and Ferreira, E.C. (2003). Estimation of multiple biomass growth rates and biomass concentration in a class of bioprocesses. *Bioproc. Biosyst. Eng.*, 25, 395–406.

Rao, C.V., Rawlings, J.B., and Mayne, D.Q. (2003). Constrained state estimation for nonlinear discrete-time systems: Stability and moving horizon approximations. *IEEE T. Automat. Contr.*, 48(2), 246–258.

Singh, A.K. and Hahn, J. (2005). Determining optimal sensor locations for state and parameter estimation for stable nonlinear systems. *Ind. Eng. Chem. Res.*, 44(15), 5645–5659.

Plantwide Control

Oral Session

Feedforward for stabilization

Morten Hovd * **Robert R. Bitmead** **

* *Engineering Cybernetics Department, Norwegian University of Science and Technology, N-7491 Trondheim, Norway (morten.hovd@itk.ntnu.no)*
** *Department of Mechanical and Aerospace Engineering, University of California San Diego, 9500 Gilman Drive, La Jolla, CA 92093-0411, USA*

Abstract: This paper demonstrates how feedforward control can assist in stabilizing unstable systems. Feedback control is necessary for stabilization, but feedforward can be used to avoid input constraints which would otherwise cause the system to go unstable. Thus, if disturbances can be measured, feedforward from disturbances can be a simple and low cost way of avoiding loss of stability due to input constraints.

Keywords: Feedforward, input constraints, stabilization

1. INTRODUCTION

Fundamental limitations in achievable control performance have received a lot of attention in the control literature. A number of important results in this area is covered in Skogestad and Postlethwaite (2005). One such fundamental limitation for unstable systems is that the range of actuation for the inputs must be sufficiently large to avoid saturation. If the inputs saturate, feedback is broken, and hence the stabilizing effect of the controller is lost. Ensuring that the inputs do not saturate is therefore important in order to guarantee closed loop stability, although an unstable system may remain stable despite the inputs being saturated for a limited period, as shown in Favez et al. (2006). If input saturation is avoided, local (linear) stability of the closed loop system is sufficient for stability.

Feedforward is normally used to improve control performance at high frequencies, beyond the achievable bandwidth for stable closed loop control. In this paper, feedforward is instead used to reduce the magnitude of the plant input moves, and therefore to avoid instability due to input constraints.

2. BACKGROUND

Consider a controlled system such as the one illustrated in Fig. 1. For the linear, unconstrained case with only feedback control ($K_f = 0$), we get

$$u = KSr - KSG_d d \tag{1}$$

where $S = (I + GK)^{-1}$. The dependence on the Laplace variable s is suppressed for notational convenience, whenever it is not needed for clarity.

Glover (1986) has shown that for unstable systems, the minimal achievable H_∞ norm of KS is given by

$$\|KS\|_\infty \geq 1/\underline{\sigma}_H(\mathcal{U}(G)^*) \tag{2}$$

where $\underline{\sigma}_H$ denotes the smallest Hankel singular value, and $\mathcal{U}(G)^*)$ denotes the anti-stable part of the plant G, with its unstable pole(s) mirrored into the left half plane.

Observe that for relationships like (2) to have any relevance for evaluating the likelihood of input saturation - with subsequent loss of stabilizing feedback - the plant model G needs to be appropriately scaled. Skogestad and Postlethwaite (2005) recommend scaling plant inputs such that $|u| < 1$ corresponds to inputs within the range of actuation, and scaling outputs such that $|y| < 1$ means that the control offset is acceptable. Similarly, the inputs of the disturbance model G_d should be scaled to get $|d| < 1$ for the expected range of disturbances, and outputs scaled in the same way as for G. In scaled variables, the references are then scaled to give $|r| < R(\omega)$ for the expected range of reference changes. Such scaling is implicitly assumed throughout this paper, and consequently the input saturation limits are assumed to be at ± 1.

Thus, with variables appropriately scaled, sinusoidal reference changes will not cause input saturation provided

$$\|KS\|_\infty < 1/R(\omega) \forall \omega \tag{3}$$

Although reference signals may contain more than a single frequency, and input saturation due to reference changes may therefore occur even if this relationship is fulfilled, this relationship is nevertheless useful in assessing whether

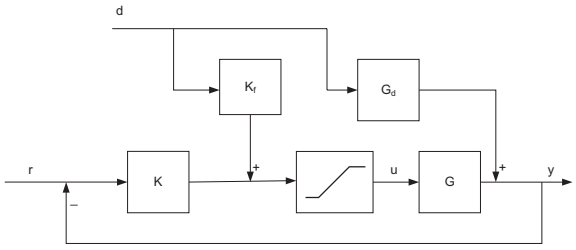


Fig. 1. Feedback and feedforward control, with limited input actuation range.

input saturation is a problem. However, it is also clear from the above that input saturation due to reference changes is not a fundamental problem - one may simply reduce the magnitude of the reference changes to avoid saturation.

On the other hand, it is typically not possible to control the magnitude of external disturbances d. Karivala et al. (2005) extended Glover's result to find that when using feedback only

$$\|KSG_d\|_\infty \geq 1/\underline{\sigma}_H(\mathcal{U}(G_{d,ms}^{-1}G)^*) \tag{4}$$

where $G_{d,ms}$ is the minimum phase and stable version of G_d, i.e., with both RHP poles and RHP zeros mirrored into the left half plane.

Accounting for the feedforward term K_f (but still assuming the saturation element to be inactive), we get

$$u = KSr + S_I(K_f - KG_d)d \tag{5}$$

where $S_I = (I + KG)^{-1}$. Note that $S = S_I$ for SISO systems, but this need not be the case for multivariable systems. From (5) we observe that introducing feedforward gives a new degree of freedom for minimizing input usage in the face of disturbances. Below, we will investigate in what situations this allows for a significant reduction of input usage, thus enabling closed loop stability.

3. STABLE DISTURBANCE MODELS

Consider the case where the plant is unstable from input to output, and hence requires feedback control for stabilization, but the unstable mode is not excited by the disturbances. This is motivated by the following example from Skogestad and Postlethwaite (2005):

Example:

$$G(s) = \frac{5}{(10s+1)(s-1)} \tag{6}$$

$$G_d(s) = \frac{k_d}{(s+1)(0.2s+1)} \tag{7}$$

The transfer functions are assumed to be appropriately scaled, as described above. From (4), we find that for $k_d > 0.54$, $\|KSG_d\|_\infty \geq 1$ for any feedback controller, and hence sinusoidal disturbances can drive the inputs to saturation. This is further illustrated in Skogestad and Postlethwaite (2005), where a stabilizing feedback controller is designed, but where saturation occurs for a step disturbance of magnitude 1 with $k_d = 0.5$. We seem to be in the paradoxical situation where control is not needed to counter the effect of disturbances (since a control offset of 1 is acceptable), but the controller needed to stabilize the system saturates due to the presence of the disturbance. Clearly, it would be better to do nothing to counteract the disturbance, but only manipulate the input to provide stabilization. However, a standard feedback controller does not distinguish the control offset caused by the (stable) disturbance from the offset caused by the unstable mode.

Equation (4) does not distinguish between stable and unstable disturbance models. For stable disturbance models, the feedforward controller K_f can be used to counter the effect of the disturbance on the input. That is, in stead of the conventional (ideal) feedforward

$$K_f = -G^{-1}G_d \tag{8}$$

which cancels the effect of the disturbance on the output [1], the ideal feedforward can from (5) be seen to be

$$K_f = KG_d \tag{9}$$

which cancels the effect of the disturbance on the input.

With this in mind, we revisit the example above, for the case with $k_d = 1$, meaning that feedback alone will not be able to maintain stability in the face of disturbances. The controller

$$K(s) = \frac{(10s+1)^2}{s(0.01s+1)} \tag{10}$$

will stabilize the unconstrained system. However, in Fig 2 we see that a unit step in the disturbance (applied at time $t = 1s$) will drive the input to saturation. Figure 3 shows that the system goes unstable as a result of the saturation. This is exactly as expected. The feedback controller $K(s)$ in (10) contains an integrator, and hence direct application of the ideal feedforward in (9) will mean that K_f will contain an integrator that is not stabilized by feedback. To avoid this problem, the controller is implemented as illustrated in Fig. 4, with the integrator in the block K_2. The corresponding feedforward is $K_f = K_1G_d$, with the overall feedback controller given by $K = K_2K_1$.

With this slight modification, we obtain the results in Figs. 5 and 6. The solid lines represents the 'ideal' feedforward control according to (9), whereas the dash-dot line is conventional feedforward according to (8). Clearly, the conventional feedforward does not avoid the input saturation. On the other hand, the modified feedforward according to (9) simply does nothing to counter the effect of the disturbance. Even though the control offset is acceptable according to the scaling used, most people would probably prefer the responses represented by the dashed line. This is obtained by augmenting the feedforward in (9) with a high pass filter, and results in offset-free control at steady state. Clearly, the pass band of the high pass filter should include frequencies significantly lower than that corresponding to the RHP pole(s).

4. UNSTABLE DISTURBANCE MODELS

It was shown above that it is simple to use feedforward from the disturbance to avoid input saturation and hence loss of stabilizing feedback, when the plant is unstable but the disturbance transfer function is stable. If the disturbance transfer function is unstable, the issue becomes more complicated.

Note, that for stabilization of the unstable disturbance transfer function to make sense, the unstable mode(s) must also be a part of the plant transfer function. That is, it must be possible to reformulate the plant and disturbance transfer functions as indicated in Fig. 7, with $G_3(s)$ a

[1] Neglecting for the moment the effects of possibly unknown initial conditions for the disturbance dynamics.

stable transfer function. In this case, the disturbance will obviously excite the unstable mode, and it therefore does not make sense to avoid the use of the manipulated input when a disturbance occurs.

Furthermore, the direct application of the 'ideal' feedforward in (9) would mean using an unstable feedforward element K_f, which would lead to an internally unstable control system. Instead, we would like to find the *stable* feedforward element K_f which minimizes the term $(K_f - KG_d)$ in (5). The term KG_d can be split into a stable and an anti-stable part. The stable part can be used directly in K_f, whereas we need a *stable approximation to the anti-stable part of* KG_d.

Approximation of an anti-stable transfer function by a stable transfer function (or *vice versa*) is known as a Nehari extension problem. That is, we want to find the optimal stable $Q(s)$ such that $\|Q(s)+R(s)\|_\infty$ is minimized, where $R(s)$ is ant-stable. A solution to this problem can be found in Glover (1984). In Glover (1984), it is also shown that the optimal error is given by $\|R^*\|_H$, where $\|\cdot\|_H$ denotes the Hankel norm, and R^* is the 'stable version of R', with the unstable poles mirrored into the left half plane. Thus, we would like to design a feedback controller K that not only stabilizes the plant, but also makes the Hankel norm of the unstable part of KG_d small. However, with

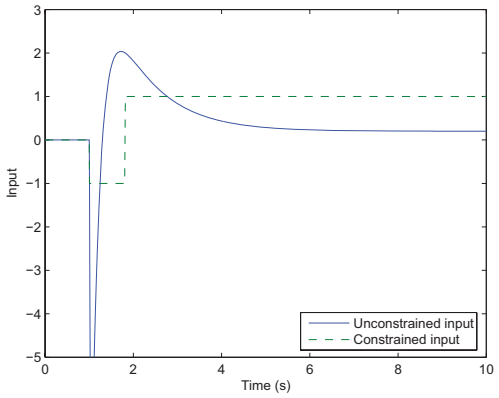


Fig. 2. Response in the input to a unit step in the disturbance as time $t = 1s$, using only feedback control.

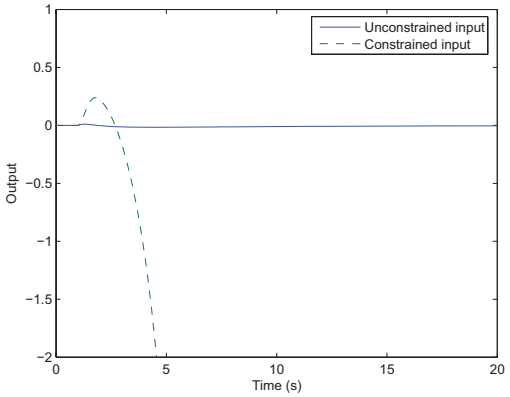


Fig. 3. Response in the output to a unit step in the disturbance as time $t = 1s$, using only feedback control.

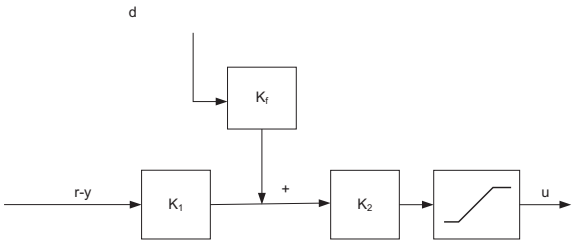


Fig. 4. Implementation of overall feedback/feedforward controller, with the integrator in the block K_2.

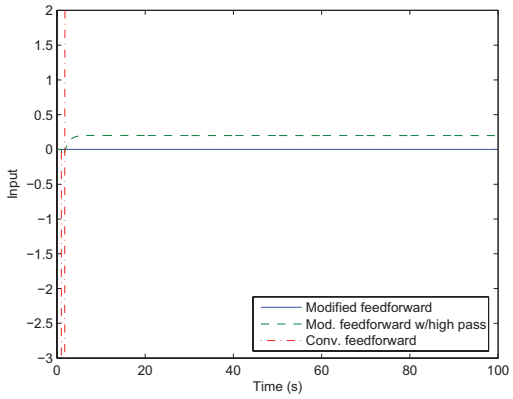


Fig. 5. Response in the input to a unit step in the disturbance as time $t = 1s$, using combined feedback and feedforward control.

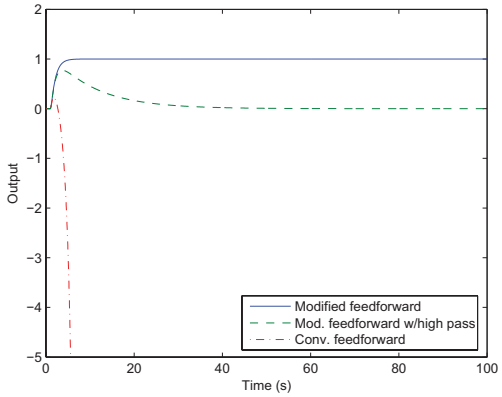


Fig. 6. Response in the output to a unit step in the disturbance as time $t = 1s$, using combined feedback and feedforward control.

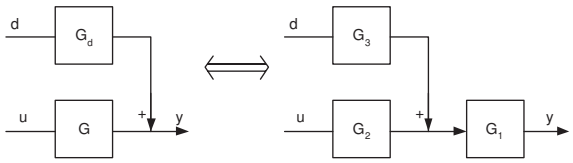


Fig. 7. Reformulation of plant and disturbance transfer functions.

a simple reformulation of the feedforward arrangement, this complication is easily avoided. This rearrangement is illustrated in Fig. 8, and may be regarded more as a 'reference governor' approach than feedforward in the ordinary sense.

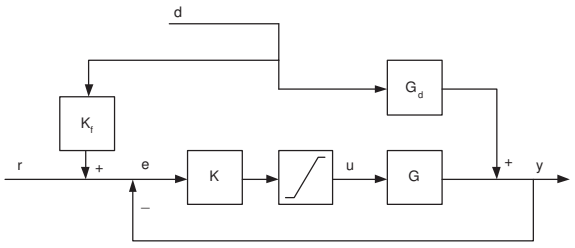


Fig. 8. 'Feedforward' arrangement for an unstable disturbance transfer function.

With the rearranged 'feedforward', the transfer function from the disturbance d to e, the input to the feedback controller K (assuming the saturation element is inactive), is given by

$$e = S(K_f - G_d)d \tag{11}$$

where $S = (I+GK)^{-1}$. Thus, for a given controller K, the controller input (and therefore also the controller output) will be small if the term $(K_f - G_d)$ is small. Next, the validity of the bound (4) and the design of the feedforward controller will be illustrated for two different cases:

- A disturbance transfer function G_d whose only non-minimum phase term is an unstable pole.
- A disturbance transfer function G_d with non-minimum phase terms in addition to the unstable pole.

The benefit of feedforward will be found to be different in these two cases. However, first the H_∞ problem formulation will be briefly explained. For the case with a stable disturbance transfer function, this was not needed, since the design of the feedforward controller was done separately from the design of the feedback controller.

4.1 H_∞ problem formulation

Using feedforward in combination with feedback means that we are using a controller with two degrees of freedom (2-DOF controller)). We wish to investigate the benefit of using a 2-DOF controller for stabilizing the system while minimizing the use of inputs in the face of measured disturbances. However, the resulting H_∞ control synthesis problem violates the standard assumptions. Assumptions A2 and A4 of Zhou et al. (1996), p. 450 are violated.

A small measurement noise n is therefore added, and the magnitude of that measurement noise is reduced until further reduction does not significantly affect the H_∞ norm achieved. The block diagram corresponding to the resulting controller synthesis problem is shown in 9.

4.2 The unstable pole as the only non-minimum phase term in G_d

The same plant transfer function as in (6) is used, whereas the disturbance transfer function is modified to

$$G_d(s) = \frac{k_d}{(s-1)(0.2s+1)} \tag{12}$$

The parameter value $k_d = 1$ is still used. First, a realization of $[G_d \quad G]$ with only one unstable mode is found. Then a 2-DOF controller is designed according to Fig. 9, and compared to a 1-DOF controller (feedback only) designed to minimize KSG_d. For both cases, a H_∞ norm of 1.83 is achieved. This also corresponds to the bound in (4). In this case, there is thus no advantage derived from using a 2-DOF controller with feedforward from disturbances [2].

However, if the feedback controller is designed for some other criterion than minimising $\|KSG_d\|_\infty$, there may be a possible advantage in designing the feedforward using the idea of approximating G_d with a stable transfer function. To illustrate, we first design a feedback controller for minimizing $\|KS\|_\infty$, achieving a H_∞ norm of 4.40 - which agrees with the bound in (2). Using this controller, we would also get $\|KSG_d\|_\infty = 4.40$. Instead, we augment the controller with feedforward as illustrated in Fig. 8. The transfer function G_d can be split into stable and unstable parts, giving

$$G_{d,stable} = \frac{-5k_d}{6(s+5)}$$
$$G_{d,unstable} = \frac{5k_d}{6(s-1)}$$

The task is this to find a stable approximation to $G_{d,unstable}$. The formulae in Glover (1984) for doing so are not directly applicable, since $G_{d,unstable}$ has only one state. However, it is easily verified that a stable approximation which achieves the minimum bound on the approximation error is given by

$$\tilde{G}_{d,unstable} = -\frac{5k_d}{12} \tag{13}$$

With the feedforward $K_f = G_{d,stable} + \tilde{G}_{d,unstable}$ used as illustrated in Fig. 8, and the feedback controller K which minimizes $\|KS\|_\infty$, we achieve an H_∞ norm of 1.83 from disturbance d to input u, while maintaining closed loop stability.

4.3 G_d with non-minimum phase terms in addition to the unstable pole

Consider next the case when the unstable disturbance transfer function is augmented with an all-pass term, giving

[2] And, in order to achieve $|u| < 1$ we would need $k_d < 0.54$, as in the original example in Skogestad and Postlethwaite (2005).

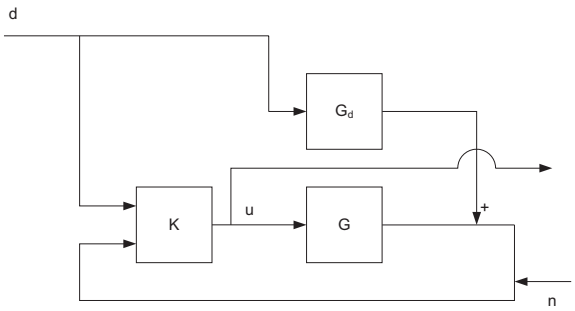


Fig. 9. H_∞ controller synthesis setup for 2-DOF controller.

$$G_d(s) = \frac{k_d(-10s+1)}{(s-1)(0.2s+1)(10s+1)} \tag{14}$$

The all-pass part of G_d cancels in the calculation of the bound in (4), and thus the minimum that can be achieved with feedback control alone is still $\|KSG_d\|_\infty = 1.83$. On the other hand, with a 2-DOF controller we achieve an H_∞ norm from d to u of 1.50. The same is achieved when designing a 1-DOF H_∞ controller minimizing $\|KS\|_\infty$ and subsequently adding feedforward to this controller in a manner similar to the preceding section.

Looking at the unstable part of $G_d(s)$ in (14), the reason for the improvement in input usage when adding the feedforward becomes apparent. One now finds that

$$G_{d,unstable} = \frac{-5k_d}{6(s-1)}\frac{9}{11}$$

The reduction in the unstable part of the disturbance transfer function by the factor 9/11 is a direct result of the all-pass term $(-10s+1)/(10s+1)$, since it modifies the residue at $s = 1$ by that same factor in the partial factor expansion of $G_d(s)$. Note that the improvement in H_∞ norm due to the introduction of feedforward also corresponds to the factor 9/11.

Stable all-pass terms will always reduce all residues in the RHP, and hence always reduce the size of the anti-stable part of $G_d(s)$ if there is a single unstable pole or a single pair of unstable complex conjugate poles. This covers a large fraction of the unstable system dynamics met in engineering practice. However, in general the unstable part of $G_d(s)$ may consist of several terms. The effect of all-pass terms will be different for the different unstable terms in $G_d(s)$, and it may therefore be possible for the unstable part of $G_d(s)$ to increase due to the presence of all-pass terms in the disturbance transfer function.

5. CONCLUSIONS

This paper illustrates how feedforward may be applied to reduce the input usage necessary for stabilizing unstable plants. If the disturbance transfer function is stable, one can thus easily remove the problem that disturbances may cause input saturation - with resulting loss of stabilizing feedback.

For the case of an unstable disturbance transfer function, it is clearly necessary to assume that the unstable mode is shared with the plant transfer function - otherwise it cannot be stabilized by feedback around the plant.

If the unstable pole is the only non-minimum phase term in the disturbance transfer function, feedforward has not been shown to improve on the optimal H_∞ norm achievable by feedback only. The bound on the H_∞ norm from d to u was found to apply for both 1-DOF and 2-DOF controllers in the example studied. However, if the 1-DOF controller is designed according to some other criterion than that of minimizing $\|KSG_d\|_\infty$, feedforward may be used to reduce the usage of inputs.

It is also found in an example that if the disturbance transfer function includes other non-minimum phase terms than the unstable pole, a 2-DOF controller can improve upon the optimal H_∞ norm achievable with feedback only. Feedforward may also in this case be added to a previously designed 1-DOF controller to reduce the usage of inputs.

Further work is necessary to quantify the optimal H_∞ norm from disturbance to plant input that is achievable when using a 2-DOF controller. Also, to simplify the analysis, the factor S has been ignored in (11), focusing instead on keeping $(K_f - G_d)$ small. Accounting for the factor S would lead to a *frequency weighted* Nehari extension problem. The possible benefit in accounting for this frequency weighting is not clear. In the examples studied, the optimal H_∞ norm for the 2-DOF problem has been achieved by appending feedforward (designed without accounting for the frequency weighting) to a 1-DOF controller design.

For the practising engineer, this paper points to the use of feedforward from disturbances to reduce input usage for stabilization. This may be an attractive alternative compared to alternative plant modifications in order to avoid input saturation (leading to loss of stabilizing feedback) in the face of disturbances.

ACKNOWLEDGEMENTS

This work was supported by the Research Council of Norway through grant no. 170636/V30.

REFERENCES

Favez, J.Y., Mullhaupt, P., Srinivasan, B., and Bonvin, D. (2006). Attractor region of planar linear systems with one unstable pole and saturated feedback. *Journal of Dynamical and Control Systems*, 12, 331–355.

Glover, K. (1984). All optimal hankel-norm approximations of linear multivariable systems and their l^∞ error bounds. *Int. J. Control*, 39, 1115–1193.

Glover, K. (1986). Robust stabilization of linear multivariable systems: relations to approximation. *Int. J. Control*, 43, 741–766.

Karivala, V., Skogestad, S., Forbes, J.F., and Meadows, E.S. (2005). Achievable input performance of linear systems under feedback control. *International Journal of Control*, 78, 1327–1341.

Skogestad, S. and Postlethwaite, I. (2005). *Multivariable Feedback Control. Analysis and Design.* John Wiley & Sons Ltd, Chichester, England.

Zhou, K., Doyle, J.C., and Glover, K. (1996). *Robust and Optimal Control*. Prentice-Hall, Upper Saddle River, NJ, USA.

Efficient Cooperative Distributed MPC using Partial Enumeration ⋆

Gabriele Pannocchia * **Stephen J. Wright** ** **Brett T. Stewart** ***
James B. Rawlings ***

* *Dip. Ing. Chim., Chim. Ind. e Sc. Mat. (DICCISM), Univ. of Pisa, Pisa, Italy (e-mail: g.pannocchia@ing.unipi.it)*
** *Computer Science Dept., Univ. of Wisconsin, Madison, WI, USA (e-mail: swright@cs.wisc.edu)*
*** *Chemical and Biological Engineering Dept., Univ. of Wisconsin, Madison, WI, USA (btstewart@wisc.edu, rawlings@engr.wisc.edu)*

Abstract: We discuss in this paper a novel and efficient implementation of distributed Model Predictive Control (MPC) systems for large-scale systems. The method is based on Partial Enumeration (PE), an approach that allows to compute the (sub)optimal solution of the Quadratic Program associated to the MPC problem by using a solution table that stores only a few most recently optimal active sets. This method is applied to the each local MPC system with significant improvements in terms of computational efficiency, and the original PE algorithm is modified to guarantee robust stability of the overall closed-loop system. We also discuss how input constraints that involve different units, e.g. on the summation of common utility consumption, can be appropriately handled. We illustrate the benefits of proposed method by means a simulated example comprising three units.

Keywords: Distributed MPC, Partial Enumeration, Explicit MPC, QP, Plant-wide Control

1. INTRODUCTION AND MOTIVATIONS

Model predictive control (MPC) is the most successful advanced control technique applied in the process industries (Qin and Badgwell, 2003), which nowadays tend to implement MPC systems in more and more plant units. Since units are often interconnected, it is clear that in some extent different MPCs may interfere, and depending on the steady-state and dynamic coupling of the units, these interactions may limit the overall achievable performance. From a pure theoretical point of view, the desire for optimality should push practitioners to implement a smaller number of (larger) MPC systems that encompass several units. From a practical point of view, however, the use of larger number of (smaller) MPC units may be preferred due to increased flexibility of the overall plant-wide control system, e.g. when one unit requires maintenance. Furthermore, depending on the size of the overall plant, a global centralized MPC system may simply be too large and too demanding in terms of computational resources. For these reasons, researchers are investigating so-called distributed MPC strategies, which aim to interconnect different MPC units with a minimal overhead structure and without increasing the complexity of the online problem solved by each MPC unit (Venkat et al., 2007; Dunbar, 2007; Rawlings and Stewart, 2008; Aske et al., 2008).

In the design of distributed MPC systems, several different "flavors" can be considered. The first one is the fully decentralized structure: each MPC unit optimizes its own objective function and no information regarding the computed input is exchanged among the MPC units. The second one is the so-called "non-cooperative" distributed MPC: the different units exchange their optimal input sequence, i.e. each MPC unit considers the other unit's planned input sequences in its optimal control problem. Both these approaches have no proven stability properties in closed-loop. In decentralized MPC the potential for instability comes first of all by the inherent model error induced by neglecting the interactions between different units. Furthermore in both decentralized and "non-cooperative" MPC structures, instability may arise because the different MPC systems optimize over different and competing objectives. When the closed-loop system is stable, "non-cooperative" MPC leads to a so-called Nash equilibrium point, which may be arbitrarily far away from the centralized optimum, also known as Pareto equilibrium point.

These issues are extensively discussed by Venkat et al. (2006a, 2007), who proposed the so-called "cooperative" distributed MPC architecture. In this distributed MPC system, each local controller optimizes, over its inputs, a common (overall) objective function and shares the computed optimal input sequence with all other controllers. As discussed by Venkat et al. (2006a,b), this scheme guarantees nominal stability, constraint satisfaction, and convergence towards the optimal centralized MPC solution, provided that no constraint involves coupling of inputs from different units.

In a recent paper (Pannocchia et al., 2007), we proposed for large-scale centralized MPC systems a novel online solution method called Partial Enumeration (PE) that allows fast evaluation of (a sub-) optimal solution of the MPC problem. Such method shares some ideas with Explicit MPC (Bemporad et al., 2002; Alessio and Bemporad, 2008; Baotic et al., 2008), which however is applicable only to small dimensional systems. In this paper, we investigate the use of PE for the solution of the local MPC problems with the aim of increasing the size and

⋆ This work was supported by National Science Foundation (Grant CTS-0456694)

complexity of problems that can be addressed efficiently by distributed MPC systems. A second objective of the present paper is to address the issue of coupled input constraints, which may limit the achievable performance of distributed MPC systems (Rawlings and Stewart, 2008).

2. COOPERATIVE MODEL PREDICTIVE CONTROL

2.1 Overall system, input constraints and local subsystems

We consider an overall time-invariant system (plant) in the discrete-time form:

$$x^+ = Ax + Bu, \qquad y = Cx\,, \tag{1}$$

in which $x \in \mathbb{R}^n$ and $x^+ \in \mathbb{R}^n$ are the state at a given time and at the successive time, respectively; $u \in \mathbb{R}^m$ is the input and $y \in \mathbb{R}^p$ is the output. Inputs are assumed to be constrained:

$$Du \le d\,, \tag{2}$$

in which the $d \in \mathbb{R}^q$ has non-negative components.

We assume that the plant is divided into $\mathcal{M}$ sub-systems (units), and each unit i has p_i outputs which are affected, in general, by *all* plant inputs. The objective is to design for each unit a Model Predictive Controller (MPC) that optimizes over a subset of inputs denoted with $u_i \in \mathbb{R}^{m_i}$, $i = 1, \ldots, \mathcal{M}$. The complementary input vector is denoted by $\bar{u}_i \in \mathbb{R}^{m-m_i}$. The subvectors u_i are *not* assumed to be disjoint. We define the selection matrices $T_i \in \mathbb{R}^{m_i \times m}$ and $\bar{T}_i \in \mathbb{R}^{(m-m_i)\times m}$ to be row submatrices of the identity, such that

$$u_i = T_i u, \qquad \bar{u}_i = \bar{T}_i u,$$

and thus

$$u = T_i' u_i + \bar{T}_i' \bar{u}_i\,.$$

Then, the dynamic evolution of each unit $i = 1, \ldots, \mathcal{M}$ can then described in the following form:

$$x_i^+ = A_i x_i + B_i u_i + \bar{B}_i \bar{u}_i, \qquad y_i = C_i x_i\,,$$

in which we distinguish the contribution of the inputs that belong to the $i-$th unit from the contribution of the other inputs. The subset of constraints in (2) that involve only u_i can be written as

$$D_i u_i \le d_i\,, \tag{3}$$

with D_i equal to the non-zero rows of (DT_i') and with d_i the corresponding elements of d. Similarly, the subset of constraints in (2) that involve $\bar{u}_i$ can be written as $\bar{D}_i \bar{u}_i \le \bar{d}_i$ with $\bar{D}_i$ equal to the non-zero rows of $(D\bar{T}_i')$ and with $\bar{d}_i$ the corresponding elements of d.

We consider the following assumptions.

Assumption 1. (Properties of subsystems). For each subsystem $i = 1, \ldots, \mathcal{M}$, the pair (A_i, C_i) is detectable, the pair (A_i, B_i) is stabilizable, and the inequality (3) represents *all* and *only* the constraints that involve elements of input vector u_i. The system from the input $\bar{u}_i$ to y_i is stable.

Remark 2. (Shared inputs). Notice that Assumption 1 admits the possibility that some inputs belong to more than one subsystem. It does require that all constraints involving any element of u_i can be written as constraints that do not involve elements of $\bar{u}_i$. Furthermore it assumes that inputs not belonging to unit i, $\bar{u}_i$, do not excite any unstable mode of A_i.

To clarify this point we present the following example.

Example 3. (Coupled constraints). Consider an overall system with four inputs and the following input constraint matrix and right-hand-side vector:

$$D = \left[\begin{array}{cc|cc} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ -1 & 0 & 0 & 0 \\ 0 & -1 & 0 & 0 \\ 0 & 0 & -1 & 0 \\ 0 & 0 & 0 & -1 \\ \hline 0 & 1 & 1 & 0 \end{array}\right], \qquad d = \begin{bmatrix} 1 \\ 1 \\ 1 \\ 1 \\ 1 \\ 1 \\ 1 \\ 1 \\ \hline 1 \end{bmatrix}.$$

The first eight rows define upper and lower bound on each input whereas the last row defines an upper bound on the sum of second and third input. Suppose that we want to design two MPCs, one of which optimizes over (u_1, u_2) whereas the other one optimizes over (u_3, u_4). Since the last constraint involve both u_2 and u_3 that, in principle, belong to different units, in order to satisfy Assumption 1, we need to include u_3 in the set of inputs for Unit 1 and u_2 in the set of inputs of Unit 2. Thus, for Unit 1 we will consider (u_1, u_2, u_3) as inputs, and for Unit 2, we will consider (u_2, u_3, u_4) as inputs.

2.2 Centralized MPC problem

To simplify the notation and given the time invariance of the system, we consider that the current decision time to be $k = 0$. Let input, state and output targets be given, and satisfy:

$$x^s = Ax^s + Bu^s, \qquad y^s = Cx^s\,.$$

Notice that such targets can be either computed by a plant-wide steady-state optimizer or as the combination of $\mathcal{M}$ local steady-state optimizers. For convenience of notation we define:

$$w = x - x^s, \qquad v = u - u^s\,.$$

We consider a finite-horizon sequence of deviation inputs $\mathbf{v} = (v(0), v(1), \ldots, v(N-1))$ and define the overall cost:

$$V(w(0), \mathbf{v}) = \frac{1}{2}\sum_{k=0}^{N-1} w(k)'Qw(k) + v(k)'Rv(k) + \frac{1}{2}w(N)'Pw(N), \qquad \text{s.t. } w^+ = Aw + Bv\,,$$

Before defining the centralized MPC optimal problem, we make the following assumptions.

Assumption 4. (Properties of overall system). The matrices Q and R are positive definite. The matrix P is the given by $P = S_s'\Pi S_s$ with Π solution to the Lyapunov equation:

$$\Pi = A_s'\Pi A_s + S_s' Q S_s\,,$$

where (A_s, S_s) come from the real Schur decomposition of A:

$$A = [S_s \;\; S_u] \begin{bmatrix} A_s & A_{su} \\ 0 & A_u \end{bmatrix} \begin{bmatrix} S_s' \\ S_u' \end{bmatrix}\,,$$

and A_s contains all stable eigenvalues of A.

The centralized MPC controller solves the following problem:

$$\mathbb{P}: \qquad \min_{\mathbf{v}} V(w(0), \mathbf{v}) \qquad \text{s.t.}$$
$$Dv \le d - Du^s, \quad S_u' w(N) = 0\,. \tag{4}$$

Remark 5. The constraint $Dv \le d - Du^s$ is equivalent to $Du \le d$. The terminal constraint $S_u' w(N) = 0$ is present only if the system is open-loop unstable (or integrating) and is needed to zero the unstable modes at the end of the horizon N. Furthermore, the cost function term $\frac{1}{2}w(N)'Pw(N)$ represents the infinite horizon cost-to-go when $v(k) = 0$ for $k \ge N$.

2.3 Distributed cooperative MPC subproblems

Let $\bar{\mathbf{v}}_i$ be a known sequence (in deviation variables) of the inputs that do not belong to Unit i, and define the control problem solved by the $i-$th MPC controller as follows:

$$\mathbb{P}_i: \quad \min_{\mathbf{v}} V(w(0), \mathbf{v}) \qquad \text{s.t.}$$
$$Dv \le d - Du^s, \quad S_u' w(N) = 0, \quad \bar{\mathbf{T}}_i \mathbf{v} = \bar{\mathbf{v}}_i \,, \quad (5)$$

in which $\bar{\mathbf{T}}_i \in \mathbb{R}^{(m-m_i)N \times mN}$ is the block diagonal matrix formed with $\bar{T}_i$, $i = 1, \ldots, \mathcal{M}$. Similarly, later we use $\mathbf{T}_i \in \mathbb{R}^{m_i N \times mN}$ to denote the block diagonal matrix formed with blocks equal to T_i. We denote with $\tilde{\mathbf{v}}_i$ the solution to (5).

Remark 6. The last equality constraint enforces the inputs that do not belong to Unit i to be equal to the known value $\bar{\mathbf{v}}_i$.

The problem $\mathbb{P}_i$ (5) contains a large number of decision variables that are fixed, namely, all inputs of the other units. We can eliminate these inputs and reformulate this problem as follows. Let the deviation input sequence $\mathbf{v}$ be expressed as

$$\mathbf{v} = \mathbf{T}_i' \mathbf{v}_i + \bar{\mathbf{T}}_i' \bar{\mathbf{v}}_i \,, \quad (6)$$

in which $\mathbf{v}_i = \mathbf{T}_i \mathbf{v}$ is the sequence of inputs that belong to Unit i, and $\bar{\mathbf{v}}_i = \bar{\mathbf{T}}_i \mathbf{v}$ is the sequence of complementary inputs. We can now write the local control problem as:

$$\mathbb{P}_i: \quad \min_{\mathbf{v}_i} V(w(0), \mathbf{v}) \qquad \text{s.t. (6) and}$$
$$D_i v_i \le d_i - D_i u_i^s, \quad S_u' w(N) = 0 \,. \quad (7)$$

Remark 7. We note that in (7) we consider only constraints for the inputs of Unit i, and constraints for the other inputs are assumed to be satisfied, i.e. $\bar{D}_i \bar{v}_i \le \bar{d}_i - \bar{D}_i \bar{u}_i^s$. Moreover, the terminal state constraint may contain equations that are not affected by $\mathbf{v}_i$, and such constraints can be eliminated.

2.4 Algorithm and properties

In distributed MPC, each local MPC unit optimizes and communicates its solution with other MPC units, forming a convex combination of the all $\mathcal{M}$ unit solutions to obtain an overall solution. If decision time permits, this procedure is repeated iteratively until convergence or until a specified maximum number of iterations is reached. The distributed MPC algorithm is initiated with an overall input sequence computed at the previous decision time, as follows:

$$\mathbf{v}^0 = (u^*(1) - u^s, \ldots, u^*(N-1) - u^s, 0) \,, \quad (8)$$

in which we emphasize that the terms $u^*(\cdot)$ are the components of the (sub)optimal sequence computed at the previous decision time, whereas u^s is the input target at the current decision time.

Remark 8. Such initial sequence is feasible with respect to the input constraint $Dv \le d - Du^s$, and it is also feasible for the terminal constraint $S_u' w(N) = 0$ if it exists, provided the target has not changed from the previous decision time.

We now describe the distributed cooperative MPC algorithm.

Algorithm 1. (Distributed Cooperative MPC). Data: current target (u^s, x^s), deviation state $w(0) = x - x^s$, an overall initial sequence $\mathbf{v}^0$ as in (8). Relative tolerance parameter ρ, maximum number of iterations $l_{\max}$.

(1) (Local MPC problems) Set $l = 1$ and for each MPC unit i repeat the following steps:
   (a) Define $\bar{\mathbf{v}}_i = \bar{\mathbf{T}}_i \mathbf{v}^{l-1}$, solve problem $\mathbb{P}_i$. Let $\mathbf{v}_i$ be the optimal solution to $\mathbb{P}_i$.
   (b) Construct the "complete" solution obtained by Unit i: $\tilde{\mathbf{v}}_i = \mathbf{T}_i' \mathbf{v}_i + \bar{\mathbf{T}}_i' \bar{\mathbf{v}}_i$.

(2) (Convex Step) Define the "overall" solution as combination of the local solutions $\mathbf{v}^l = \sum_{i=1}^{\mathcal{M}} \lambda_i \tilde{\mathbf{v}}_i$, with $\lambda_i > 0$ and $\sum_{i=1}^{\mathcal{M}} \lambda_i = 1$.

(3) (Convergence Test) If $\frac{\|\mathbf{v}^l - \mathbf{v}^{l-1}\|}{1 + \|\mathbf{v}^{l-1}\|} < \rho$ or $l = l_{\max}$, set $\mathbf{v}^* = \mathbf{v}^l$ and stop. Otherwise, increase $l \leftarrow l + 1$ and go to 1.

It is possible to show that such cooperative MPC algorithm converges to the optimal centralized solution in the limit of infinite iterations. Furthermore, we can establish closed-loop stability for any finite number of iterations l.

3. PARTIAL ENUMERATION

3.1 Introduction

Both the centralized problem $\mathbb{P}$ and each problem $\mathbb{P}_i$ can be written as convex Quadratic Programs, and for small to medium scale systems, the solution can be computed efficiently using either Active Set Method (ASM) or Interior Point Method (IPM) solvers (Rao et al., 1998; Bartlett et al., 2002; Milman and Davison, 2003). However, as the system dimension increases, online solvers cannot provide a solution within an acceptable decision time. In order to compute a (suboptimal) solution for large-scale systems that are currently out of the range of QP solvers, we recently proposed an approach called Partial Enumeration (Pannocchia et al., 2007). In Partial Enumeration (PE) we use a solution table that stores a (small) number of optimal active sets and the associated piecewise linear solution (Bemporad et al., 2002). This approach was applied to large-scale centralized MPC problems in (Pannocchia et al., 2007) with average speed-up factors of 80-200 times compared to conventional QP solvers, and with small closed-loop suboptimality. We review PE here and make appropriate modifications for applying it to the distributed MPC problem $\mathbb{P}_i$.

3.2 PE algorithm and properties

We first consider the centralized MPC problem $\mathbb{P}$ and write it as a parametric QP as follows:

$$\min_{\mathbf{v}} \frac{1}{2} \mathbf{v}' \mathbf{H} \mathbf{v} + \mathbf{v}' \mathbf{G} w(0) + \frac{1}{2} w(0)' \mathbf{P} w(0) \qquad \text{s.t.} \quad (9a)$$
$$\mathbf{D}\mathbf{v} + \mathbf{C} u^s \le \mathbf{d}, \quad \mathbf{E}\mathbf{v} + \mathbf{F} w(0) = 0. \quad (9b)$$

Note that $z = [w(0)', \ u^{s\prime}]'$ is the parameter that changes at each decision time point, while all other terms are constant and omitted in the sake of space.

Given a point $\mathbf{v}^*$, we denote by $(\mathbf{D}_a, \mathbf{C}_a, \mathbf{d}_a)$ the stacked rows of $(\mathbf{D}, \mathbf{C}, \mathbf{d})$ such that $\mathbf{D}_a \mathbf{v}^* + \mathbf{C}_a u^s = \mathbf{d}_a$ (i.e. the active constraints). We also denote with $(\bar{\mathbf{D}}_a, \bar{\mathbf{C}}_a, \bar{\mathbf{d}}_a)$ the complementary stacked rows, i.e. such that $\bar{\mathbf{D}}_a \mathbf{v}^* + \bar{\mathbf{C}}_a u^s < \bar{\mathbf{d}}_a$ (i.e. the inactive constraints). Next, we define:

$$\mathcal{G} = [\mathbf{G} \ 0] \,, \quad \mathcal{A} = \begin{bmatrix} \mathbf{D}_a \\ \mathbf{E} \end{bmatrix}, \quad \mathcal{B} = \begin{bmatrix} 0 & \mathbf{C}_a \\ \mathbf{F} & 0 \end{bmatrix}, \quad \mathbf{b} = \begin{bmatrix} \mathbf{d}_a \\ 0 \end{bmatrix}.$$

In order for $\mathbf{v}^*$ to be optimal for (9), the following first-order optimality KKT conditions must hold:

$$\mathbf{H}\mathbf{v}^* + \mathcal{G} z + \mathcal{A}' \lambda^* = 0, \quad (10a)$$
$$\mathcal{A}\mathbf{v}^* + \mathcal{B} z = \mathbf{b}, \quad (10b)$$
$$\lambda_j^* \ge 0, \qquad j \in \{\text{indices of active inequalities}\}, \quad (10c)$$
$$\bar{\mathbf{D}}_a \mathbf{v}^* + [0 \ \bar{\mathbf{C}}_i] z \le \bar{\mathbf{d}}_a. \quad (10d)$$

We now solve the system (10) to derive $\mathbf{v}^*$ as a linear function of the parameter z and we derive the conditions on z for which the considered active set is optimal. To this aim, several

approaches can be followed, and in this paper we use the so-called Null-Space method.

Let $\mathcal{Z}$ be a full rank matrix such $\mathcal{A}\mathcal{Z} = 0$, and consider the point $\mathbf{v}_0 = \mathcal{A}^+(\mathbf{b} - \mathcal{B}z)$, which $\mathcal{A}^+$ is the pseudo-inverse of $\mathcal{A}$. We can express any point that is feasible for (10b) as $\mathbf{v} = \mathbf{v}_0 + \mathcal{Z}p$ and thus rewrite (10a) as follows:

$$\mathbf{H}\mathcal{Z}p + \mathbf{H}\mathbf{v}_0 + \mathcal{G}z + \mathcal{A}'\lambda^* = 0\,.$$

Next, we multiply (on the left) by $\mathcal{Z}'$ (to eliminate the term $\mathcal{Z}'\mathcal{A}'\lambda^*$) and solve for p to obtain

$$\begin{aligned} p &= \left(\mathcal{H}^{-1}(\mathcal{Z}'\mathbf{H}\mathcal{A}^+\mathcal{B} - \mathcal{Z}'\mathcal{G})\right) z - \mathcal{H}^{-1}(\mathcal{Z}'\mathbf{H}\mathcal{A}^+)\mathbf{b} \\ &= \mathbf{\Gamma}_p z + \gamma_p\,, \end{aligned}$$

with $\mathcal{H} = \mathcal{Z}'\mathbf{H}\mathcal{Z}$. Finally, we compute $\mathbf{v}^*$ as follows:

$$\begin{aligned} \mathbf{v}^* &= \mathbf{v}_0 + \mathcal{Z}p = \mathcal{A}^+(\mathbf{b} - \mathcal{B}z) + \mathcal{Z}(\mathbf{\Gamma}_p z + \gamma_p) \\ &= \mathbf{\Gamma} z + \gamma\,. \end{aligned} \tag{11}$$

Now (10a) can be solved for λ^* as follows:

$$\lambda^* = -(\mathcal{A}')^+(\mathbf{H}\mathbf{v}^* + \mathcal{G}z) = -(\mathcal{A}')^+(\mathbf{H}\mathbf{\Gamma} + \mathcal{G})z - (\mathcal{A}')^+\mathbf{H}\gamma.$$

Finally, we write the Primal and Dual inequalities (10c) and (10d) as follows:

$$\begin{bmatrix} \bar{\mathbf{D}}_a\mathbf{\Gamma} + [0\ \bar{\mathbf{C}}_a] \\ [I\ 0]\,(\mathcal{A}')^+(\mathbf{H}\mathbf{\Gamma} + \mathcal{G}) \end{bmatrix} z \leq \begin{bmatrix} \bar{\mathbf{d}}_a - \bar{\mathbf{D}}_a\gamma \\ -\,[I\ 0]\,(\mathcal{A}')^+\mathbf{H}\gamma \end{bmatrix},$$

or more concisely as

$$\begin{bmatrix} \mathbf{\Psi}_P \\ \mathbf{\Psi}_D \end{bmatrix} z \leq \begin{bmatrix} \psi_P \\ \psi_D \end{bmatrix}. \tag{12}$$

Furthermore, by inserting the solution (11) into the objective function of (9), we can write the optimal cost for the current active set as: $V^*(z) = \frac{1}{2}z'V_2 z + V_1 z + V_0$.

In Partial Enumeration we store $(\mathbf{\Psi}_P, \mathbf{\Psi}_D, \mathbf{\Gamma})$, (ψ_P, ψ_D, γ), and also (V_0, V_1, V_2), for a fixed number of active sets that were optimal in the most recent decision time points. Online we scan the table to check if, for the given parameter z, optimality conditions (12) are satisfied, and in such case compute the optimal solution from (11). However, given the fact that not all possible optimal active sets are stored, it is possible that no entry in the table is optimal. In such cases it is necessary to compute a suboptimal solution for closed-loop control. Nonetheless, a QP solver is called afterwards to compute the optimal solution $\mathbf{v}^*$, and thus derive the matrices/vectors $(\mathbf{\Psi}_P, \mathbf{\Psi}_D, \mathbf{\Gamma})$, (ψ_P, ψ_D, γ), (V_0, V_1, V_2) for the corresponding optimal active sets. Whenever this table entry becomes available, it is inserted into the table. When the table exceeds its maximum size (defined by the user), we delete the entry that was optimal least recently. Thus, the table size is fixed and hence the table lookup process is fast, but the table entries are updated to keep track of new operating conditions for the plant.

In order to compute a suboptimal input sequence when the table does include the optimal active set for the current parameter z several options can be considered. It is important to ensure that the given suboptimal solution guarantees, at least, nominal closed-loop stability, and this can be obtained if we ensure a cost decrease from the previous decision time point. Here, we propose a procedure that allows us to prove robust stability of the closed-loop under PE MPC. The procedure requires two points, the first one which needs to be feasible and its computation is discussed later in Algorithm 2. The second point, instead, is a particular minimizer of (9a) subject to the equality constraint (if present) and all the input inequalities that are active at the target point. More specifically, given the input target u^s, let $(\bar{\mathbf{D}}, \bar{\mathbf{d}})$ denote the subset of rows of $(\mathbf{D}, \mathbf{d})$ such that $\bar{\mathbf{C}}u^s = \bar{\mathbf{d}}$. We define $\hat{\mathbf{v}}$ as the solution to:

$$\min_{\mathbf{v}} V(w(0), \mathbf{v}) \text{ s.t. } \bar{\mathbf{D}}\mathbf{v} = 0,\ \mathbf{E}\mathbf{v} + \mathbf{F}w(0) = 0\,. \tag{13}$$

We can show that $\hat{\mathbf{v}} = \hat{\mathbf{\Gamma}}(u^s)w(0)$, where the dependence of the matrix $\hat{\mathbf{\Gamma}}$ on u^s comes from the fact that u^s defines $(\bar{D}, \bar{d})$.

In the following, we denote by $\mathbf{v}^0 = (u^*(1) - u^s, \ldots, u^*(N-1) - u^s, 0)$ the previous shifted optimal sequence, where the inputs $(u(1)^*, \ldots, u^*(N-1))$ were computed at the previous decision time, while u^s is the current input target.

Algorithm 2. (Partial Enumeration). Data: table with M entries, each comprising the terms $(\mathbf{\Psi}_P, \mathbf{\Psi}_D, \mathbf{\Gamma})$, (ψ_P, ψ_D, γ), (V_0, V_1, V_2); current parameter $z = [w(0)', u^{s\prime}]'$; candidate sequence $\mathbf{v}^0$ and its cost V^0 if feasible (otherwise $V^0 = \infty$); maximum table size $M_{\max}$. Output: Input sequence $\mathbf{v}^*$ and updated table. Set $j = 0$, $\tilde{V} = V^0$, $\tilde{\mathbf{v}} = \mathbf{v}^0$.

(1) (Table scanning.) Set $j \leftarrow j+1$. If $j > M$ and $\tilde{\mathbf{v}}$ is feasible go to 4. If $j > M$ and $\tilde{\mathbf{v}}$ is infeasible go to 3. Otherwise, perform the following steps for the $j-$th entry:
   (a) If $\mathbf{\Psi}_P z \leq \psi_P$ does not hold, go to 1. Otherwise,
   (b) If $\mathbf{\Psi}_D z \leq \psi_D$ holds go to 2. Otherwise,
   (c) Compute the cost V. If $V < \tilde{V}$, set $\tilde{\mathbf{v}} = \mathbf{\Gamma}_u(\mathbf{d}_a - \mathbf{C}_a u^s) + \mathbf{\Gamma}_w w(0)$. Go to 1.

(2) (Optimal solution found.) Compute the optimal solution $\mathbf{v}^*$. Inject the optimal input. Put this entry in first position of the table. Stop.

(3) (Feasibility recovery; arrive at this step only if $\tilde{\mathbf{v}}$ is not feasible.) Solve the LP

$$\min_{\mathbf{q},\mathbf{s}} \mathbf{1}'(\mathbf{q} + \mathbf{s}) \qquad \text{s.t. } \mathbf{D}(\mathbf{q} - \mathbf{s}) \leq \mathbf{r}_1,$$
$$\mathbf{E}(\mathbf{q} - \mathbf{s}) = \mathbf{r}_2, \quad \mathbf{q} \geq 0, \quad \mathbf{s} \geq 0$$

   where $\mathbf{r}_1 = \mathbf{d} - \mathbf{C}u^s - \mathbf{D}\tilde{\mathbf{v}}$, $\mathbf{r}_2 = -\mathbf{F}w(0) - \mathbf{E}\tilde{\mathbf{v}}$, and $\mathbf{1}$ is the vector of ones. Redefine $\tilde{\mathbf{v}} \leftarrow \tilde{\mathbf{v}} + \mathbf{q} - \mathbf{s}$ and compute its cost $\tilde{V}$.

(4) (Solution improvement; $\tilde{\mathbf{v}}$ is feasible at this point.) Evaluate $\hat{\mathbf{v}}$, and compute the largest $t \in [0, 1]$ such that $\mathbf{D}(\hat{\mathbf{v}} - \tilde{\mathbf{v}})t \leq \mathbf{d} - \mathbf{C}u^s - \mathbf{D}\tilde{\mathbf{v}}$. Set $\mathbf{v}^* = \tilde{\mathbf{v}}(1-t) + t\hat{\mathbf{v}}$.

(5) (Table update, performed in parallel.) Solve the QP (9), and find the terms $(\mathbf{\Psi}_P, \mathbf{\Psi}_D, \mathbf{\Gamma})$, (ψ_P, ψ_D, γ), (V_0, V_1, V_2) for the optimal active set. Insert this entry in first position of the table, set $M \leftarrow M+1$. If $M = M_{\max}+1$, delete the entry that was optimal least recently, and set $M = M_{\max}$.

Remark 9. The "feasibility recovery" step 3 is required *only if* the system is open-loop unstable *and* either the target changed from the previous decision time or a disturbance occurred. In the nominal case without target change, such step is not performed because $\mathbf{v}^0$ is always feasible. Step 3 is the only "expensive" computation in Algorithm 2 and is justified by closed-loop stability reasons of an open-loop unstable plant. For input bound constraints (i.e., $u_{\min} \leq u \leq u_{\max}$) further simplifications that allow increased speedup and lower sub optimality can be considered.

It can be shown that PE MPC is nominally stabilizing and robustly stabilizing for sufficiently small measurement noise and additive disturbances (Pannocchia et al., 2009).

3.3 Application of Partial Enumeration to cooperative MPC

Each $\mathbb{P}_i$ in (7) can be written as the following parametric QP:

Table 1. Outputs and inputs of the three units, according to two design schemes: Design A (existing), Design B (optimal).

	Outputs	Inputs	
		Design A	Design B
Unit 1	(y_1, y_2, y_3)	(u_1, u_2, u_3)	$(u_1, u_2, u_3, u_4, u_8)$
Unit 2	(y_4, y_5, y_6)	(u_4, u_5, u_6)	$(u_3, u_4, u_5, u_6, u_8)$
Unit 3	(y_7, y_8)	(u_7, u_8)	(u_3, u_4, u_7, u_8)

$$\min_{\mathbf{v}_i} \frac{1}{2}\mathbf{v}_i'\mathbf{H}_i\mathbf{v}_i + \mathbf{v}_i'\mathcal{G}_i z_i + \frac{1}{2}z_i\mathbf{P}_i z_i \qquad \text{s.t.} \tag{14a}$$

$$\mathbf{D}_i\mathbf{v}_i + \mathbf{C}_i z_i \leq \mathbf{d}, \quad \mathbf{E}_i\mathbf{v}_i + \mathbf{F}_i z_i = 0\,, \tag{14b}$$

in which $z_i = [z',\ \bar{\mathbf{v}}_i']'$ is the parameter augmented with the sequence of inputs that do not belong to Unit i, and $\mathbf{H}_i = \mathbf{T}_i\mathbf{H}\mathbf{T}_i'$, $\mathcal{G}_i = \left[\mathcal{G}\ \mathbf{T}_i\mathbf{H}\bar{\mathbf{T}}_i'\right]$, $\mathbf{P}_i = \begin{bmatrix} \mathbf{P} & 0 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} \mathcal{G}'\bar{\mathbf{T}}_i' \\ \bar{\mathbf{T}}_i\mathcal{G}\ \ \bar{\mathbf{T}}_i\mathbf{H}\bar{\mathbf{T}}_i' \end{bmatrix}$, $\mathbf{D}_i = \mathbf{D}\mathbf{T}_i'$, $\mathbf{C}_i = \left[0\ \mathbf{C}\ \mathbf{D}\bar{\mathbf{T}}_i'\right]$, $\mathbf{E}_i = \mathbf{E}\mathbf{T}_i'$, $\mathbf{F}_i = \left[\mathbf{F}\ 0\ \mathbf{E}\bar{\mathbf{T}}_i'\right]$. Notice that several rows of $\mathbf{D}_i$ and $\mathbf{E}_i$ are zero and can be deleted along with the corresponding rows of $\mathbf{C}_i$ and $\mathbf{F}_i$.

We notice that the QP (14) is in the same form of (9), with the main difference that the parameter z is augmented with the known sequence of inputs not belonging to Unit i. Given this increase in dimensionality, a full explicit MPC is impractical even for small systems. On the other hand, PE Algorithm 2 can be readily applied to solve (14). Since PE does not guarantee that each $\mathbb{P}_i$ is solved exactly, no convergence to the optimal centralized solution can be proved. Nonetheless, we can show closed-loop nominal stability and robust stability for sufficiently small disturbances.

4. APPLICATION EXAMPLE

4.1 Overall system and units definition

As an example, we consider a stable system with 8 inputs, 8 outputs and 48 states, whose details are omitted in the sake of space. Each input of the system is constrained in $[-1, 1]$, and the following coupled constraint holds:

$$[0\ 0\ 0\ 1\ 1\ 0\ 0\ 1]\, u \leq 1\,. \tag{15}$$

In the MPC design we use: $Q = I$, $R = I$, and $N = 30$.

We consider that this overall plant is divided in three units. Outputs and inputs of each unit are reported in Table 1, where we emphasize two different design schemes. In Design A, which can be regarded as the existing scheme for this plant, no inputs belong to more than one unit at a time. However, because of the coupled constraint (15), such input partition scheme does not satisfy Assumption 1. Therefore, for such scheme convergence to the optimal centralized solution cannot be guaranteed. For this reason, we consider an alternative input partition scheme (Design B) in which the inputs (u_3, u_4, u_8) belong to all three units.

4.2 Effect of coupled constraints

First of all we investigate about the different convergence properties for the two distributed MPC architectures. We consider that at decision time 10, the input target changes from 0 to $u^s = (0, 0, 0.5, 0.2, 0, 0, 0, 0.3)'$, thus making the coupled constraint active. We report in Figure 1 the closed-loop response of $u_3 + u_4 + u_8$ obtained by three controllers: CMPC is the centralized controller, DMPC-A is the distributed control structure with $l = 1$ iteration based on Design A, DMPC-B is the distributed control structure with $l = 1$ iteration based on Design B. For this study, we solve the optimal control problems exactly, i.e. we do not use Partial Enumeration. We report in Table 2, the suboptimality of DMPC-A and DMPC-B as the number of iterations l increases, defined by the index:

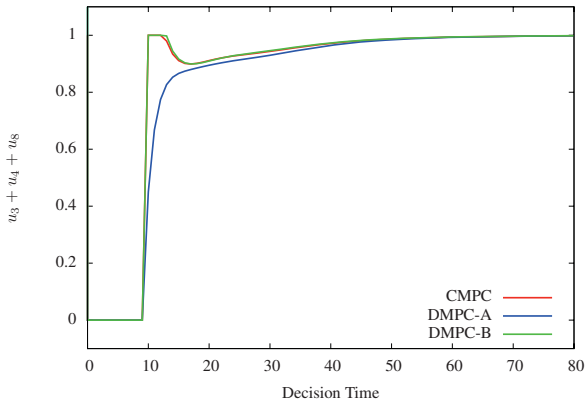


Fig. 1. Effect of coupled input constraints: closed-loop response of $u_3 + u_4 + u_8$ for centralized MPC (CMPC), distributed cooperative MPC based on Design A (DMPC-A), distributed cooperative MPC based on Design B (DMPC-B). Both DMPC-A and DMPC-B make $l = 1$ iteration.

Table 2. Suboptimality of DMPC-A and DMPC-B for different number of iterations l.

Dec. MPC	S_I			
	$l = 1$	$l = 5$	$l = 10$	$l = 50$
DMPC-A	22.9	0.885	0.682	0.673
DMPC-B	2.29	$4.86 \cdot 10^{-2}$	$5.43 \cdot 10^{-4}$	$1.53 \cdot 10^{-8}$

$$S_I = 100\frac{V_{CL} - V_{CL}^*}{V_{CL}}\,,$$

in which V_{CL} is the closed-loop cost for the considered (distributed) controller and V_{CL}^* is the closed-loop cost for the optimal centralized controller. As expected DMPC-B handles the coupled constraint much better than DMPC-A, and as the number of iterations increases, DMPC-B converges to the optimal centralized MPC solution, whereas the suboptimality index for DMPC-A does not go to zero.

4.3 Comparison of PE-based and QP-based distributed MPC

We now present the results for several decentralized controllers that solve the local MPC problems $\mathbb{P}_i$ either via PE or via an exact (active set) QP solver. We are interested in assessing the suboptimality of each scheme, as well as the computational efficiency quantified by the two indices [1] :

- Average Speed Factor: $A_{SF} = \frac{T_{\text{aver}}^*}{T_{\text{aver}}}$ where T_{aver}^* is the average CPU time required to solve the centralized problem $\mathbb{P}$ via QP solver, and T_{aver} is the average CPU time required to compute the solution using Algorithm 1 (either via PE or via exact QP solver).
- Worst Case Speed Factor: $W_{SF} = \frac{T_{\max}^*}{T_{\max}}$, where $T_{\max}^*$ and $T_{\max}$ are the maximum CPU times for the centralized (QP based) problem $\mathbb{P}$ and for the distributed Algorithm 1 (PE or QP based), respectively.

[1] All computations are performed using GNU Octave on a Pentium-M (1.86 GHz, 1 GB RAM) running Linux.

Table 3. Comparison of suboptimality and computational efficiency for several DMPC-B, based on PE or exact QP solver

	QP based			PE based		
Iter.	S_I	A_{SF}	W_{SF}	S_I	A_{SF}	W_{SF}
$l = 1$	14.7	5.24	12.1	14.8	93.5	285
$l = 5$	0.604	1.25	4.66	0.612	22.0	66.7
$l = 10$	0.0408	0.797	2.37	0.0490	13.9	34.4

We consider a closed-loop simulation of 5000 decision time points, in the presence of random output noise, affecting the state estimate and the target at each decision time, and 14 large target changes. When PE is used, each MPC unit deploys an initially empty table of maximum dimension $M_{\max} = 10$. The results are summarized in Table 3. We can observe, first of all, that as number of iterations l increases, DMPC-B converges to the centralized optimum performance, as indicated by the negligible suboptimality index S_I. Next, we can see that for a given number of iterations l, the suboptimality index obtained by solving the local problems with the QP solver is essentially equal to that obtained with the PE solver. However, the computational requirements using QP and PE solvers are remarkably different. If we compare the distributed controllers using the same number of iterations, DMPC-B based on local PE solvers can compute the solution 17–18 times faster (on average), 14–24 times faster (worst case) than the corresponding DMPC-B based on local QP solvers. In practice, since the time allowed for computation of local solutions and iterations among the distributed controllers may be regarded as fixed, the goal of using local PE solvers is that we can allow more iterations and thus (almost) achieve the centralized optimal performance. If compared with the centralized MPC, most of the computational benefits of using DMPC-B based on PE solvers are achieved with a limited number of iterations, e.g. $l = 5$, which allows one to obtain a suboptimality less than 1% with an average speedup factor of 22 and a worst case speedup factor of 67.

A final remark can be made regarding the possible (apparent) overlap of applicability and scope of Partial Enumeration and distributed MPC, i.e. as alternative means for solving MPC problems in large-scale systems. We want to stress that the main motivation for distributed MPC is organizational rather than computational and, in fact, if the number of iterations l is increased the distributed MPC architecture (based on QP solvers) may be even more time consuming than a centralized MPC architecture (notice the average "speedup" factor less than 1 for DMPC-B based on QP with $l = 10$ iterations). Therefore, distributed MPC should not be considered as a possible competitor of Partial Enumeration centralized MPC which, on the other hand, is motivated by computational issues.

5. CONCLUSIONS

We proposed in this paper an efficient implementation for distributed cooperative Model Predictive Control. The approach is based on Partial Enumeration, that solves the Quadratic Program associated to the MPC problem by means of a small solution table, which includes the most recently optimal active sets. If the optimal solution is not found in the table, a quick suboptimal solution is computed for closed-loop control. In parallel, the optimal active set is evaluated and inserted into the table, possibly deleting the least recently optimal active set. In this way the size of the table is kept small, thus limiting the required time for scanning it. We applied such approach for the solution of "local" MPC problems that are solved in each unit of a distributed MPC system. We also revised the cooperative distributed MPC architecture to optimally handle the case of coupled input constraints. Finally, we presented a simulation example of an 8 input 8 output plant comprising three units in which we achieved relevant speedup factors and negligible suboptimality compared to QP-based MPC.

REFERENCES

Alessio, A. and Bemporad, A. (2008). A survey on explicit model predictive control. In *Proceedings of the International Workshop on Assessment and Future Directions of NMPC*. Pavia, Italy.

Aske, E.M.B., Strand, S., and Skogestad, S. (2008). Coordinator mpc for maximizing plant throughput. *Comp. Chem. Eng.*, 32, 195–204.

Baotic, M., Borrelli, F., Bemporad, A., and Morari, M. (2008). Efficient on-line computation of constrained optimal control. *SIAM J. Control Optim.*, 47, 2470–2489.

Bartlett, R.A., Biegler, L.T., Backstrom, J., and Gopal, V. (2002). Quadratic programming algorithms for large-scale model predictive control. *Journal of Process Control*, 12(7), 775–795.

Bemporad, A., Morari, M., Dua, V., and Pistikopoulos, E.N. (2002). The explicit linear quadratic regulator for constrained systems. *Automatica*, 38, 3–20.

Dunbar, W.B. (2007). Distributed receding horizon control of dynamically coupled nonlinear systems. *IEEE Trans. Auto. Contr.*, 52, 1249–1263.

Milman, R. and Davison, E.J. (2003). Fast computation of the quadratic programming subproblem in model predictive control. In *Proceedings of the American Control Conference*, 4723–4729.

Pannocchia, G., Rawlings, J.B., and Wright, S.J. (2007). Fast, large-scale model predictive control by partial enumeration. *Automatica*, 43(5), 852–860.

Pannocchia, G., Wright, S.J., and Rawlings, J.B. (2009). On the robust stability of partial enumeration model predictive control. *In preparation*.

Qin, S.J. and Badgwell, T.A. (2003). A survey of industrial model predictive control technology. *Contr. Eng. Pract.*, 11, 733–764.

Rao, C., Wright, S.J., and Rawlings, J.B. (1998). Application of interior-point methods to model predictive control. *J. Optim. Theory Applic.*, 99, 723–757.

Rawlings, J.B. and Stewart, B.T. (2008). Coordinating multiple optimization-based controllers: New opportunities and challanges. *J. Proc. Contr.*, 18, 839–845.

Venkat, A.N., Rawlings, J.B., and Wright, S.J. (2006a). Stability and optimality of distributed, linear MPC. Part 1: state feedback. Technical Report 2006–03, TWMCC, Department of Chemical and Biological Engineering, University of Wisconsin–Madison (Available at http://jbrwww.che.wisc.edu/tech-reports.html).

Venkat, A.N., Rawlings, J.B., and Wright, S.J. (2006b). Stability and optimality of distributed, linear MPC. Part 2: output feedback. Technical Report 2006–04, TWMCC, Department of Chemical and Biological Engineering, University of Wisconsin–Madison (Available at http://jbrwww.che.wisc.edu/tech-reports.html).

Venkat, A.N., Rawlings, J.B., and Wright, S.J. (2007). Distributed model predictive control of large-scale systems. In *Assessment and Future Directions of Nonlinear Model Predictive Control*, 591–605. Springer.

Optimality of Process Networks. ⋆

Michael R. Wartmann * **B. Erik Ydstie** *

* *Department of Chemical Engineering, Carnegie Mellon University, Pittsburgh, USA (e-mail: wartmann@cmu.edu, ydstie@cmu.edu).*

Abstract: In this paper we show that conservation laws for extensive quantities and the second law of thermodynamics lead to conditions for stability and optimality of a process network. Interconnections among nodes are represented through connectivity matrices and network graphs. A generalized version of Tellegen's theorem from electrical circuit theory plays a central role in deriving the objective function of the regarded dynamic process networks. The application of irreversible thermodynamics lead to stability and optimality results based on the co-content and content of the regarded process networks. The principle is illustrated in a pipeflow example.

Keywords: dissipation, network theory, irreversible thermodynamics, distributed control, passivity theory.

1. INTRODUCTION

The complexity of process systems arises from the variety of how simple subunits are connected (Hangos et al. (1999)). A crucial component in modeling process systems is therefore to understand how connections between the subunits lead to complex system behavior. Ydstie and Alonso (1997) developed a theoretical framework providing a link between passivity theory and physics using the second law of thermodynamics. They discussed the need to develop passivity based control techniques which focus on input-output properties of the systems. An understanding for complex behavior can then be derived from macroscopic thermodynamic constraints instead of microscopic equations and the complexity that results from using very detailed models can be reduced. Jillson and Ydstie (2007) developed a topological result similar to Tellegen's theorem of electrical circuit theory and passivity theory to derive sufficient conditions under which a network is stabilized using decentralized feedback. The theory shows that it is possible to control very complex networks of process systems without actually modeling the thermodynamics and kinetics explicitly. This is due to inherent passivity properties that follow from the second law of thermodynamics. The conditions for passivity can be checked in a distributed manner. In this work, we will explore if similar ideas can be applied for optimization.

We extend the approaches in Ydstie and Alonso (1997); Jillson and Ydstie (2007) to provide an organizational framework for treating complex process systems concerning optimality using ideas from network theory. The formalism of network theory has been particularly successful for modeling and control of dynamic systems in electrical engineering applications. Classically, electrical circuit theory is not considered an application of non-equilibrium thermodynamics. Nevertheless, electrical circuits are typical irreversible thermodynamic systems. The formalism developed in electrical circuit theory was extended to general thermodynamic systems by Oster et al. (1971); Peusner (1986). In particular the application to complex biological systems has been carried out successfully by Oster and Desoer (1971); Mickulecky (2001). In this paper, we apply the formalism of network theory to describe connected process systems. Network theory brings thermodynamics a degree of mathematical rigor and allows to unify ideas from non-equilibrium thermodynamics, dynamic system theory and control. In the context of dissipativity of process systems, network theory facilitates the extension of irreversible thermodynamics by the system's topological description which is an important part of the dynamic behavior. Looking at mathematical models of many dynamic physical systems, we can identify a certain inherent structure. We can separate the network model into a kinematic structure which addresses the topology of the system and a dynamical structure (Oster and Desoer (1971)). The connectivity properties of the system describe the physical processes where the dynamical structure defines the relationships between the state variables. The paper is organized as follows: In Section 2, we define the type of process systems and describe the connection to network theory, in Section 3, we describe fundamental topological properties of the regarded process networks. In Section 4 and 5, we elaborate the concepts of stability and optimality for the regarded systems and present a pipeflow example to illustrate our findings in Section 6.

2. PROCESS NETWORKS

Process networks are written as a collection of interconnected sub-systems

$$\dot{x}_i = F(x_i) + \sum_{j=0, j\neq i}^{n} G(u_i, x_i, x_j), \quad i = 0, ..., n \tag{1}$$

$$y_i = H(x_i) \tag{2}$$

x_i is the state of subsystem i and $x_i(0)$ is the initial condition. The function F describes the unforced motion of the system, the function G describes how the system is

⋆ This work was supported by the Center for Integrated Operations in the Petroleum Industry, Trondheim, Norway.

connected with other sub-systems, and the output function H relates the state of the system to the measurement functions y_i. The functions u_i represent the manipulated variables. The functions F, G, H are all differentiable at least once. The state of the entire network is given by the vector $x = (x_0^T, x_1^T, ..., x_n^T)^T$.

Subscript zero refers to the reference (exo-) system. Often we are not interested in the dynamics of the exo-system, or more likely, it is too complex to model. The process system is modeled as the reduced system without the reference sub-system. Its state is given by the vector $x = (x_1^T, ..., x_n^T)^T$. The interactions with the exo-system are then established through the boundary conditions.

The network form, as illustrated in Figure 1 is convenient when we model systems with a graph structure. In such systems the interactions between the sub-systems depend on the state of the sub-system itself and the state of its immediate neighbors. Not all dynamical systems can be decomposed in this fashion. However, many large scale systems have sparse interconnections and they can be modeled compactly as networks of sub-systems with interconnections. It is also easy to see that many physical systems, especially those that satisfy the principle of local action, can be decomposed in the manner shown in (1).

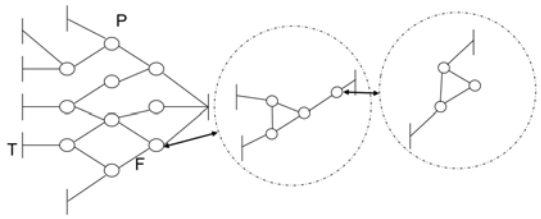


Fig. 1. Graphical network representation: Topological structure of a network consisting of nodes, terminals, and flows. Nodes can contain subgraphs and give rise to a hierarchical, multiscale structure.

We define the inventory Z of a sub-system or a group of systems to be a non-negative, additive function of the state of the corresponding sub-system(s). By additivity we mean that if Z_1 is the inventory of sub-system 1 and Z_2 is the inventory of sub-system 2, then $Z_1 + Z_2$ is the total inventory. Hence for any i, j

$$Z\begin{pmatrix} x_i \\ x_j \end{pmatrix} = Z(x_i) + Z(x_j)$$

By non-negativity we mean that the inventory cannot be less than zero. Examples of physical inventories include mass, energy and charge. More generally, an inventory is any property which is related to an amount.

By referring to (1) and using continuity we derive the conservation law

$$\frac{dZ_i}{dt} = p_i(x_i) + \sum_{j=1, j\neq i}^{n} f_{ij}(u) \tag{3}$$

The *drift* $p_i(x_i) = \frac{\partial Z(x_i)}{\partial x_i} F(x_i)$ measures the rate of production and the function $f_{ij}(u) = \frac{\partial Z(x_i)}{\partial x_i} G(u, x_i, x_j)$ measures the supply of Z between sub-systems j and i. We have the symmetry condition

$$f_{ij}(u) = f_{ji}(u)$$

The term

$$\phi(u, z, d) = \sum_{j=1, j\neq i}^{n} -f_{ij}(u)$$

therefore measures the net rate of supply to sub-system i from all other sub-systems. It is called the *action* on sub-system i.

Definition: Let X_0 be a subset of state-space. An inventory defined by (3) is said to have the

(1) *Clausius-Planck property* if $p(x) > 0$ for x not in X_0
(2) *Conservation property* if $p(x) = 0$ for all x not in X_0
(3) *Dissipation inequality* if $p(x) < 0$ for x not in X_0

The set X_0 associated with the dissipative action ϕ is called the set of *passive states*.

By a graph $\mathbf{G}$ we mean a finite set $\upsilon(\mathbf{G}) = (\upsilon_1, .., \upsilon_{n_P})$, whose elements are called **nodes**, together with the set $\epsilon(\mathbf{G}) \subset \upsilon \times \upsilon$, whose elements are called **branches**. A branch is therefore an ordered pair of distinct nodes.

- If, for all $(\upsilon_i, \upsilon_j) \in \epsilon(\mathbf{G})$, the branch $(\upsilon_j, \upsilon_i) \in \epsilon(\mathbf{G})$ then the graph is said to be **undirected**. Otherwise, it is called a **directed graph**.
- A branch (υ_i, υ_j) is said to be **incoming with respect to** υ_j and **outgoing with respect to** υ_i and can be represented as an arrow with node υ_i as its tail and node υ_j as its head.

Definition 1. A network of nodes $P_i, i = 1, ..., n_p, n_p + 1, ..., n_v$ consisting of nodes and terminals interconnected through branches $E_i, i = 1, ..., n_f$ with topology defined by the graph

$$\mathbf{G} = (\mathbf{E}, \mathbf{P})$$

is called a *process network* if its interconnection structure is described by a directed graph and we have

(1) **First law:** There exists an inventory E (the energy) which satisfies the conservation property
(2) **Second law:** There exists an inventory S (the entropy) which satisfies the Clausius-Planck property

We now develop a compact description of the topology of the network by introducing the incidence matrix.

Definition 2. The $n_t \times n_f$ matrix $\mathbf{A_a}$ is called incidence matrix for the matrix elements a_{ij} being

$$a_{ij} = \begin{cases} 1, & \text{if flow } j \text{ leaves node } i \\ -1, & \text{if flow } j \text{ enters node } i \\ 0, & \text{if flow } j \text{ is not incident with node } i \end{cases}$$

One node of the network is set as reference or datum node P_0 representing the exo-system. The $(n_t - 1) \times n_f$ matrix $\mathbf{A}$, where the row that contains the elements a_{0j} of the reference node P_0 is eliminated, is called reduced incidence matrix.

The connections between nodes through branches can be uniquely defined using the incident matrix $\mathbf{A}$. The conservation laws (3) can now be written

$$\mathbf{AF} = \mathbf{0} \tag{4}$$

for the node-to-branch incident matrix $\mathbf{A}$, where $\mathbf{F^T} = [\frac{dZ_1}{dt}, \frac{dZ_2}{dt}, .., \frac{dZ_t}{dt}, f_{12}, f_{13}..f_{n_{t-1}, n_t}, p_1, .., p_t]$. The flows f_{ij} represent connections between two nodes i.e. f_{ij} connects node i to node j, p_i denotes sources or sinks. The direction of the flows are defined according to the directionality

established in the graph. We now define a vector $\mathbf{W}$ so that

$$\mathbf{W} = \mathbf{A^T w} \tag{5}$$

where $\mathbf{W}$ are the potential differences across flow connections. The variables w are conjugate to Z if they are related via the Legendre transform of a convex potential like the entropy.

A dual structural representation can be derived using mesh analysis (the analysis developed above, which is based on the conservation laws, is called node analysis). Mesh analysis is counter-intuitive in process control applications but frequently used for electrical circuit analysis. When introducing stability and optimality concepts in this work, we will focus on describing the primal problem and its implications but refer to the dual mesh-based problem as proving the equivalent dual case.

2.1 Constitutive Relations

Constitutive equations relate efforts and flows (resistive), flows and displacements (capacitive), and efforts and fluxes (inductive). The constitutive equations describe energy dissipating, irreversible processes (resistive) or energy storing, reversible processes. The constitutive equations define the type of energetic transaction inside the process system or between the process system and the environment. The three main types can be described as

- Capacitive constitutive equation: storage of potential energy, $W = f_C(Z)$
- Inductive constitutive equation: storage of kinetic energy, $p = f_L(W)$
- Resistive constitutive equation: dissipation of energy, $F = f_R(W)$

In the context of process networks, storage of energy usually occurs through capacitive elements. In this work, inductive constitutive equations are neglected due to the fact that we focus on chemical processes or chemical process plants in which inertial effects in mass flow and thus accumulation of kinetic energy are not a significant contributor to the energy balance.

3. TOPOLOGICAL RESULTS, CONTENT AND CO-CONTENT

Consider two networks (a) and (b) with the same topology (identical incidence matrix) but not necessarily the same state. Denote the variables in network (a) with the superscript a and denote variables in the other network with superscript b. Using the conservation laws (4) we can then write $\mathbf{W^{a}}^T\mathbf{F^b} = (\mathbf{A}^T\mathbf{w^a})^T\mathbf{F^b} = \mathbf{w^a}^T\mathbf{A}\mathbf{F^b} = 0$. The equality

$$\mathbf{W^{a}}^T\mathbf{F^b} = 0$$

is often called *Tellegen's theorem.*

Without the reference system Tellegen's theorem is written

$$\mathbf{w^b}^T\frac{d\mathbf{Z^a}}{dt} = -\mathbf{W_R^b}^T\mathbf{F_R^a} - \mathbf{w_T^b}^T\mathbf{F_T^a} - \mathbf{w^b}^T\mathbf{p^a} \tag{6}$$

The term of the left hand side is called the storage. The three terms on the right refer to power dissipation due to transportation, supply from the exo-system through the terminals, and dissipation by production respectively.

If we consider a single network ($a = b$), then we can drop the superscript and we get the common form $\mathbf{W}^T\mathbf{F} = 0$ which represents a powerbalance. If Z represents the energy then we get the classical energy balance for the network. The "balance of entropy dissipation" results if we let one inventory correspond to the thermodynamic entropy defined so that

$$S = k_B \ln \Omega(x)$$

It is important to note that the fundamental equation gives a definition of the classical entropy in terms of extensive and intensive variables through the Pfaffian

$$TdS = dU + PdV - \sum_{i=1}^{n_c} \mu_i dN_i$$

Tellegen's theorem applied to the primal (extensive variable) vector $Z = (U, V, N_1, .., N_n)$ and its Legendre dual (intensive variable) vector $w = (1, P, \mu_1, .., \mu_n)/T$ then gives

$$\dot{S} = \mathbf{W_R}^T\mathbf{F_R} + \mathbf{w_T}^T\mathbf{F_T} + \mathbf{w}^T\mathbf{p}$$

where we used the fundamental equation and (6). The term $p_S = \mathbf{F_R}^T\mathbf{W_R} + \mathbf{w^T p}$ is called the rate of entropy generation.

We define the content of the network as the integral

$$G_R = \int_0^{\mathbf{F_R}} \mathbf{W_R}^T d\mathbf{F_R} + \int_0^{\mathbf{p}} \mathbf{w}^T d\mathbf{p} \tag{7}$$

The co-content is given as

$$G_R^* = \int_0^{\mathbf{W_R}} \mathbf{F_R}^T d\mathbf{W} + \int_0^{\mathbf{w}} \mathbf{p}^T d\mathbf{w} \tag{8}$$

By integration by parts and proper choice of the constant p^* (the constant of integration) we see that

$$G_R + G_R^* = p_S \geq 0$$

The second law dictates that the inequality holds (positive entropy production).

4. STABILITY OF PROCESS NETWORKS

In this section we derive a stability result using a combination of Tellegen's theorem and the co-content as a line integral. First, we note that Tellegen's theorem shows that for each time t the vectors $\mathbf{W}$ and $\mathbf{F}$ lie in fixed and orthogonal spaces. The identity $\dot{\mathbf{W}}^T\mathbf{F} = 0$ is therefore valid for all t and by taking out the sub-system which represent the exo-system we can write as before

$$\dot{\mathbf{Z}}^T\dot{\mathbf{w}} = -\mathbf{F_R^T}\dot{\mathbf{W}}_\mathbf{R} - \mathbf{F_T^T}\dot{\mathbf{w}}_\mathbf{T} - \mathbf{p_T^T}\dot{\mathbf{w}} \tag{9}$$

Due to the concavity of the entropy function we know that there exists a matrix $\mathbf{M} \geq 0$ so that $d\mathbf{w} = \mathbf{M}d\mathbf{Z}$, hence

$$\dot{\mathbf{Z}}^T\dot{\mathbf{w}} = \dot{\mathbf{Z}}^T M\dot{\mathbf{Z}} \geq 0$$

We can also write the co-content as a line integral

$$G_R^* = \int^t (\mathbf{F_R}^T\dot{\mathbf{W}}_\mathbf{R} + \mathbf{p}^T\dot{\mathbf{w}})dt \geq 0$$

Hence, by integrating (9) we get

$$\int_0^t \dot{\mathbf{Z}}^T M\dot{\mathbf{Z}}dt = -G_R^* - \mathbf{F_T^T}\dot{\mathbf{w}}_\mathbf{T}$$

The contribution due the terminal potentials vanish if $\dot{\mathbf{w}}_\mathbf{T} = 0$ and it follows that G^* is integrable and subject the condition of uniform continuity we conclude that G^* converges which implies that $\dot{\mathbf{w}}$ converges to zero.

5. OPTIMALITY OF PROCESS NETWORKS

Maxwell (1892) formulated the minimum heat theorem which states that for linear resistive electrical circuits driven by constant power sources, the flows distribute themselves in a way as to minimize the heat that is dissipated through the resistive elements. Prigogine (1947) observed that the theorem can be generalized to thermodynamic systems with the entropy production σ_S being minimized at steady state. Based on Tellegen's theorem and the content and co-content, we can propose an optimization problem that allows us to find the steady state and dynamic trajectory of a dynamic process network.

For a process network with a graph $\mathbf{G}$, we can define the extended content

$$G = \sum_{i=1}^{b} \int^{F_i} W_i dF_i = \int^{\mathbf{F}} \mathbf{W}^T d\mathbf{F} \tag{10}$$

and the extended co-content:

$$G^* = \sum_{i=1}^{b} \int^{W_i} F_i dW_i = \int^{\mathbf{W}} \mathbf{F}^T d\mathbf{W} \tag{11}$$

The extended content G and co-content G^* represent the sum of contents and co-contents for all branches i.e. reversible, irreversible, production and terminal flow connections of the network.

Lemma 3. For the network content G and the co-content G^*

$$G^*(\mathbf{W}) = \mathbf{W}^T\mathbf{F} - G(\mathbf{F}) \tag{12}$$

Equation (12) is a special form of Tellegen's theorem and can be used to do a variable change corresponding to a Legendre transformation.

Proof. The relation follows directly from integration by parts.

Lemma 4. For the sum of extended content $G = \int^{\mathbf{F}} \mathbf{W}^T d\mathbf{F}$ and extended co-content $G^* = \int^{\mathbf{W}} \mathbf{F}^T d\mathbf{W}$, the following relation holds:

$$G + G^* = 0$$

Proof. Using Tellegen's theorem and Lemma 3, the result follows immediately.

Definition 5. The following set of equations defines the process system:

$$\mathbf{AF} = \mathbf{0} \tag{13}$$
$$\mathbf{W} = \mathbf{A^T w} \tag{14}$$
$$\mathbf{F_R} = \Lambda(\mathbf{W_R}) \tag{15}$$
$$\mathbf{Z} = \mathbf{C w_C} \tag{16}$$
$$\mathbf{F_R} = \mathbf{F} - \mathbf{F_S} \tag{17}$$
$$\mathbf{W_R} = \mathbf{W} - \mathbf{W_S} \tag{18}$$
$$\mathbf{F_S} = \mathbf{F_T} \tag{19}$$
$$\mathbf{W_S} = \mathbf{W_T} \tag{20}$$
$$\mathbf{Z(0)} = \mathbf{Z_0} \tag{21}$$

The first two equations (13) and (14) are the Kirchhoff relations for process networks. Equations (15) are the resistive constitutive equations with Λ being a matrix function and (16) are the capacitive constitutive equations.

We introduced the variables $\mathbf{F_R}$ and $\mathbf{W_R}$ which facilitate writing the resistive constitutive equations in a compact way. The variables $\mathbf{F_R}$ and $\mathbf{W_R}$ allow us to include the terminals as sources or sinks through (19) and (20) for both, terminals where we have the function of the flows $\mathbf{F_T}$ or the potentials $\mathbf{W_T}$ given. For simplicity, we assume the terminal conditions as constant over time. The last equation (21) constitutes the initial conditions for the inventories $\mathbf{Z}$.

The set of equations can be transformed into a system of nonlinear differential algebraic equations (DAE) of the form

$$\frac{d\mathbf{Z}}{dt} = \mathbf{A}(\mathbf{Z}) + \mathbf{B}^{\mathbf{Z}}_{\mathbf{F}}(\mathbf{F}^{\mathbf{input}}_{\mathbf{T}}) + \mathbf{B}^{\mathbf{Z}}_{\mathbf{W}}(\mathbf{W}^{\mathbf{input}}_{\mathbf{T}}) \tag{22}$$
$$\mathbf{W}^{\mathbf{output}}_{\mathbf{T}} = \mathbf{C}^{\mathbf{W}}(\mathbf{Z}) + \mathbf{D}^{\mathbf{W}}_{\mathbf{F}}(\mathbf{F}^{\mathbf{input}}_{\mathbf{T}}) + \mathbf{D}^{\mathbf{W}}_{\mathbf{W}}(\mathbf{W}^{\mathbf{input}}_{\mathbf{T}}) \tag{23}$$
$$\mathbf{F}^{\mathbf{output}}_{\mathbf{T}} = \mathbf{C}^{\mathbf{F}}(\mathbf{Z}) + \mathbf{D}^{\mathbf{F}}_{\mathbf{W}}(\mathbf{W}^{\mathbf{input}}_{\mathbf{T}}) + \mathbf{D}^{\mathbf{F}}_{\mathbf{F}}(\mathbf{F}^{\mathbf{input}}_{\mathbf{T}}) \tag{24}$$

where nonlinearities are introduced through the constitutive equations. In this dynamic system, each terminal has an input and an output variable. The set of differential equations (22) determines the trajectories of $\mathbf{Z}$ and represent a state space system. The algebraic constraints (23) and (24) compute the output variables at the terminals from the input variables and the state $\mathbf{Z}$.

To find the stationary solutions of the system, we need to solve the set of equations

$$\mathbf{AF} = \mathbf{0} \tag{25}$$
$$\mathbf{W} = \mathbf{A^T w} \tag{26}$$
$$\mathbf{F} - \mathbf{F_T} = \Lambda(\mathbf{W} - \mathbf{W_T}) \tag{27}$$

with the three main sets of constraints: Conservations laws, uniqueness conditions, and the constitutive equations. The inventories and capacitive constitutive equations are only relevant for the dynamic case.

In the following theorem, we introduce the connection between content, co-content and the Kirchhoff laws, and present how duality of the free variables plays a crucial role for the optimization problem that is solved when a process network converges to a steady state solution. The constitutive equations are not directly involved as they are not relevant for the topological properties of the process network.

Theorem 6. For the optimization problem

$$\min_{\mathbf{w}} G^* = \int_0^W \mathbf{F}^T d\mathbf{W} \tag{28}$$
$$s.t. \quad \mathbf{W} = \mathbf{A^T w} \tag{29}$$
$$\mathbf{F} = \Lambda(\mathbf{W}) \tag{30}$$

with the cocontent G^* as objective function, the uniqueness conditions, and resistive constitutive equations as constraints, the solution exhibits a set of equations consisting of the uniqueness condition, the conservation laws, and the constitutive equations. The Langrange multipliers of the optimization problem are the network flow variables $\mathbf{F}$.

Proof. Starting with equations (28) - (30), we first substitute the constitutive equations (30) into the objective function (28) to eliminate the flow variables $\mathbf{F}$. The Lagrange function of the resulting optimization problem is

$$\min L(\mathbf{W}, \mathbf{w}, \lambda) = \int_0^W \Lambda(\mathbf{W})^T d\mathbf{W} + \lambda^T(\mathbf{A^T w} - \mathbf{W}) \quad (31)$$

First order conditions:

$$\frac{\partial L}{\partial \mathbf{W}} = \Lambda(\mathbf{W}) - \lambda = \mathbf{0} \quad (32)$$

$$\frac{\partial L}{\partial \mathbf{w}} = \mathbf{A}\lambda = \mathbf{0} \quad (33)$$

$$\frac{\partial L}{\partial \lambda} = \mathbf{A^T w} - \mathbf{W} = \mathbf{0} \quad (34)$$

comparing (32) and the constitutive equations (30), it follows that $\lambda = \mathbf{F}$. Using $\lambda = \mathbf{F}$ in (32) and (33), the result follows.

In principle, an optimization problem is solved where one set of Kirchhoff equations is omitted. Through the first order conditions, the missing set of equations is derived. The optimization problem with the Kirchhoff voltage law as constraints can be converted to an optimization problem with the Kirchhoff current law and vice versa.

We can now propose the main theorem which allows us to connect the steady state of a process network to the objective function that is simultaneously optimized i.e. we can find the natural optimization problem that a process network solves, when converging to a steady state. We explored the structure of the problem in the previous theorem, however, we need to be able to define boundary conditions and a solution for process networks connected to an exo-system.

Theorem 7. Consider a process network G with given resistive constitutive equations $\mathbf{F_R} = \Lambda(\mathbf{W_R})$ and boundary conditions for each terminal as well as one set of either the conservation laws or the uniqueness conditions. The stationary solution ($\frac{dZ_i}{dt} = 0$) for the network with conservation laws (13) and the uniqueness conditions (14)

$$\mathbf{AF} = \mathbf{0} \quad (35)$$

$$\mathbf{W} = \mathbf{A^T w} \quad (36)$$

$$\mathbf{F} - \mathbf{F_T} = \Lambda(\mathbf{W} - \mathbf{W_T}) \quad (37)$$

can be found by solving the following optimization problem

$$\min_{\mathbf{w}} \quad G^* = \int_0^W \mathbf{F}^T d\mathbf{W} \quad (38)$$

$$s.t. \quad \mathbf{W} = \mathbf{A^T w} \quad (39)$$

$$\mathbf{F_R} = \Lambda(\mathbf{W_R}) \quad (40)$$

$$\mathbf{F_T} = \text{const and/or } \mathbf{W_T} = \text{const} \quad (41)$$

or its equivalent dual optimization problem where (36) is replaced by (35).

Proof. Starting with (38) - (41), we substitute the boundary conditions (41) into the constitutive equations (40) and the constitutive equation into the objective function (38). We then form the Lagrange function using the flows $\mathbf{F}$ as Lagrange multipliers

$$L(\mathbf{W}, \mathbf{w}, \mathbf{F}) = \int_0^W (\Lambda(\mathbf{W} - \mathbf{W_T})^T d\mathbf{W} + \mathbf{W}^T \mathbf{F_T} \quad (42)$$

$$+ \mathbf{F}^T(\mathbf{A^T w} - \mathbf{W}) \quad (43)$$

First order conditions:

$$\frac{\partial L}{\partial \mathbf{W}} = \Lambda(\mathbf{W} - \mathbf{W_T}) + \mathbf{F_T} - \mathbf{F} = \mathbf{0} \quad (44)$$

$$\frac{\partial L}{\partial \mathbf{w}} = \mathbf{AF} = \mathbf{0} \quad (45)$$

$$\frac{\partial L}{\partial \mathbf{F}} = \mathbf{A^T w} - \mathbf{W} = \mathbf{0} \quad (46)$$

Comparing (25) - (27) to (44) - (46) shows the result. Concerning the second order conditions, we observe that convexity of the constraints is trivial for the linear Kirchhoff laws. Non-convexities of the optimization problem are due to non-linearities of the constitutive equations i.e. the constitutive equations are non-positive. For the second order conditions, it is apparent that the first derivative of the constitutive equations has to be analyzed and found positive definite for a global minimum, which corresponds exactly to the findings for passivity in Jillson and Ydstie (2007) for a unique network solution and convergence.

Generally, the objective function is a measure for dissipation of the storage variable over time. We conclude that the steady state of a passive network minimizes the dissipated power subject to the constraints imposed by the constitutive equations, topology, and boundary conditions, i.e. terminal connections.

6. PIPEFLOW NETWORK

A pipeline network example shows how optimization and dynamic simulation are connected. The network consists of two connected pipelines where each pipeline flows through a cylindrical storage tank with volume V_j open to the atmosphere, as shown in Fig. 2. A reference node is introduced representing the environment and connected to the terminals and dynamic nodes.

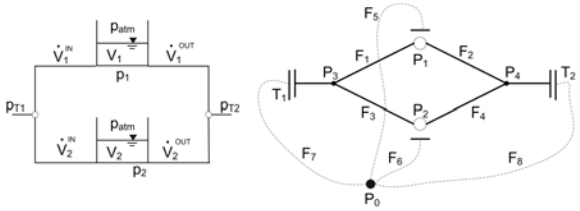


Fig. 2. Graphical network representations: Problem specific representation on the left, a generalized representation on the right including P_0 representing the exo-system.

Each pipeline's cross section is cylindrical with area A_i. The pipeline flow is given as a lumped parameter representation introducing pressure potentials p_j at the nodes and assuming laminar flow ($Re < 2300$). It is assumed that the fluid shows Newtonian behavior as well as being incompressible ($\rho = \text{const.}$). Therefore, the relation between

volumetric flow $\dot{V}_i$ and pressure drop $\Delta p_i = p_j - p_{j+1}$ can be modeled using Hagen-Poiseuille's law $\dot{V}_i = \frac{\pi r_i^4}{8\eta L_i}\Delta p_i$, where r_i is the radius of the pipeline's cross-section and L_i is the length of pipeline i. The potential at the bottom of the tank is given as $p_j = \rho g h_j + p_{atm}$ by hydrostatics. The fluid volume V_j in the tank is connected to the level h_j through $V_j = A_j h_j$ where A_j is the cross-section of the tank. We complete the model with the conservation laws for mass or, for constant density, the conservation of volume:

$$dV_1/dt = \dot{V}_1^{IN} - \dot{V}_1^{OUT} \tag{47}$$
$$dV_2/dt = \dot{V}_2^{IN} - \dot{V}_2^{OUT} \tag{48}$$
$$\dot{V}_{T1} = \dot{V}_2^{IN} + \dot{V}_2^{IN} \tag{49}$$
$$\dot{V}_{T2} = \dot{V}_2^{OUT} + \dot{V}_2^{OUT} \tag{50}$$

Initial conditions for the tank volumes $V_{0,i}$ have to be specified as well as boundary conditions at the terminals. The steady state of (47) - (50) can be found by integrating the differential equations.

The dynamic system given by the previous equations converges to the solution of the following optimization problem ($\frac{dV_1}{dt} = \frac{dV_2}{dt} = 0$):

$$\min \quad \sum_{i=1}^{4} \int_0^{\Delta p_i} \dot{V}_i d(\Delta p_i) \tag{51}$$
$$s.t. \quad (47) - (50) \tag{52}$$
$$\dot{V}_i = \frac{\pi r_i^4}{8\eta L_i}\Delta p_i \ , i = 1, .., 4 \tag{53}$$
$$\dot{V}_{T1} = const., p_{T2} = const. \tag{54}$$

Solving the optimization problem therefore corresponds to minimizing the power dissipated through viscous friction in the pipes subject to the conservation laws and boundary conditions. For each terminal, one boundary condition has to be specified which can be chosen freely ($\dot{V}_{T1} = 0.3$ m^3/s, $p_{T2} = 1.013$ bar). The parameters are given as $d = 0.5$ m and $L_1 = 2500$ m for the upper pipeline segments and $L_2 = 5000$ m for the lower segments. The tanks' cross-sectional diameter is chosen as $d_1 = d_1 = 2$ m. Fig. 3 shows the simulation results. We chose the initial conditions for $V_{0,1} = V_{0,2} = 25$ m ^{3}.

It is apparent that the value of the objective function as well as the flows of the dynamic simulation converge to the optimum determined through the optimization problem for arbitrary initial conditions. The constant inflow $\dot{V}_{T1}$ into the network divides itself into flows through the upper segments and lower segments choosing the path of least resistance.

7. CONCLUSIONS AND DISCUSSION

We introduced a new framework for analysis of optimality and stability of networked process systems in this work. We provide a systematic approach to define stability and optimality conditions for these systems. The objective function minimized by a process systems in its steady state is derived. Although for simplicity, we regard only the steady state in this example, the optimization problem

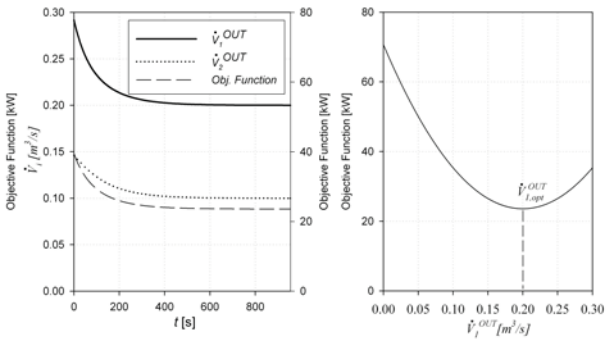


Fig. 3. Flows between tanks and outgoing terminal T_2 and the power dissipation (objective function) as a function of time on the left. Convergence of $\dot{V}_1^{OUT} =$ to 0.2 m^3/s and $\dot{V}_2^{OUT} =$ to 0.1 m^3/s. Objective function values of $\dot{V}_1^{OUT}$ on the right.

is also valid for transient conditions. The findings can explored to design decentralized control structures and hence shape the natural objective function of a process systems towards an economic objective.

REFERENCES

Hangos, K., Alonso, A., Perkins, J., and Ydstie, B. (1999). Thermodynamic approach to the structural stability of process plants. *AICHE*, 45(4), 802–816.

Jillson, K. and Ydstie, B. (2007). Process networks with decentralized inventory and flow control. *Journal of Process Control*, 17, 399413.

Maxwell, J. (1892). *A Treatise on Electricity and Magnetism*. Oxford University Press.

Mickulecky, D. (2001). Network thermodynamics and complexity: a transition to relational systems theory. *Computers and Chemistry*, 25, 369391.

Oster, G. and Desoer, C. (1971). Tellegen's theorem and thermodynamic inequalities. *J. of theor. Biology*, 32, 219–241.

Oster, G., Perelson, A., and Katchalsky, A. (1971). Network thermodynamics. *Nature*, 234, 393–399.

Peusner, L. (1986). *Studies in Network Thermodynamics*. Elsevier, Amsterdam.

Prigogine, I. (1947). *Etude thermodynamique des phenomenes irreversibles*. PhD thesis, Liege.

Ydstie, B. and Alonso, A. (1997). Process systems and passivity via the clausius-planck inequality. *Systems & Control Letters*, 30, 253–264.

Quasi-decentralized Scheduled Output Feedback Control of Process Systems Using Wireless Sensor Networks ⋆

Yulei Sun and Nael H. El-Farra *

* *Department of Chemical Engineering & Materials Science, University of California, Davis, CA 95616 USA (e-mail: nhelfarra@ucdavis.edu)*

Abstract: This paper presents a quasi-decentralized output feedback control structure for multi-unit plants with limited state measurements and distributed control systems that exchange information over a resource-constrained wireless sensor network (WSN). The networked control structure brings together model-based feedback control, state estimation and sensor scheduling to enforce closed-loop stability while simultaneously minimizing the rate of communication over the WSN. Initially, an observer-based output feedback controller is designed for each unit. To conserve the resources of the wireless devices, communication between the local control systems is suspended periodically for extended time periods during which each control system relies on models of the plant units to generate the necessary control action. Communication is then re-established at discrete time instances according to a certain schedule that determines the order and times at which the wireless sensor suites transmit the state estimates needed to update the states of the models embedded in the target units. By analyzing the combined discrete-continuous behavior of the scheduled closed-loop plant, we explicitly characterize the stability of the networked closed-loop system in terms of the communication rate, the sensor transmission schedule, the accuracy of the models, as well as the controller and observer design parameters. The results are illustrated using a chemical plant example where it is shown that by judicious management of the interplays between the control, communication and scheduling design parameters, it is possible to stabilize the plant while simultaneously enhancing the savings in WSN resources beyond what is possible with concurrent transmission configurations.

Keywords: Quasi-decentralized control, wireless sensor networks, model-based control, state estimation, scheduling algorithms, chemical plants.

1. INTRODUCTION

Chemical plants are large-scale dynamical systems that consist of a large number of distributed units which are tightly interconnected through mass and energy flows and recycle. Traditionally, the controller synthesis problem for such plants has been addressed within either the centralized or decentralized control frameworks. Both approaches have been the subject of numerous research studies aimed at understanding their advantages and limitations, as well as the development of strategies to overcome some of those limitations (e.g., see Siljak (1991); Lunze (1992); Sourlas and Manousiouthakis (1995); Katebi and Johnson (1997); Cui and Jacobsen (2002); Camponogara et al. (2002); Huang and Huang (2004); Skogestad (2004); Venkat et al. (2005); Goodwin et al. (2005); Kariwala (2007) and the references therein). Other notable contributions on this problem include the development of plant-wide control strategies based on passivity theory and concepts from thermodynamics (Hangos et al. (1999); Antelo et al. (2007)), the development of agent-based systems to control spatially-distributed reactor networks (Tetiker et al. (2008)), and the analysis and control of integrated process networks using time-scale decomposition and singular perturbations (Baldea et al. (2006)).

An approach that provides a compromise between the complexity of traditional centralized control schemes, on the one hand, and the performance limitations of decentralized control approaches, on the other, is quasi-decentralized control, which refers to a control strategy in which most signals used for control are collected and processed locally, while some signals are transferred between the local units and controllers to adequately account for the interactions and minimize the propagation of process upsets from one unit to another. A key consideration in the design and implementation of quasi-decentralized control systems is the selection of the communication medium over which the local control systems must communicate. While dedicated point-to-point links offer a reliable communication medium, the complexity and costs of installation and maintenance associated with this architecture, as well as the lack of flexibility for real-time reconfiguration, represent major drawbacks especially for large-scale plants with complex interconnections. An alternative solution is the use of wireless communication networks. The viability of this approach stems from the convergence of recent advances in actuator/sensor manufacturing, wireless communications and digital electronics,

⋆ Financial support by NSF CAREER Award, CBET-0747954, is gratefully acknowledged.

which has produced low-cost wireless sensors (e.g., Kumar (2001); Song et al. (2006)) that can be installed for a fraction of the cost of wired devices. Wireless sensor networks (WSNs) offer unprecedented flexibility ranging from high-density sensing capabilities to deployment in areas where wired devices may be difficult or impossible to deploy. Augmenting existing process control and monitoring systems with WSNs has the potential to expand the capabilities of the existing control technology beyond what is feasible with wired architectures alone. These are appealing goals that coincide with the recent calls for expanding the traditional process control and operations paradigm in the direction of smart plant operations (e.g., see Ydstie (2002); Christofides et al. (2007)).

One of the main challenges to be addressed when deploying a low-cost WSN for control is that of handling the inherent constraints on network resources, including the limitations on the computation, processing and communication capabilities. In an effort to address this problem, we developed in Sun and El-Farra (2008a) a quasi-decentralized model-based networked control architecture that enforces closed-loop stability with minimal cross communication between the constituent subsystems. The minimum allowable communication rate was characterized in terms of the plant-models' mismatch for the case when all sensors suites transmit their measurements concurrently and are given simultaneous access to the network. The networked control structure was subsequently generalized in Sun and El-Farra (2008b) to address the problem when only limited state measurements are available (for additional results and references on the design of networked control systems, the reader may refer to Walsh and Ye (2001); Montestruque and Antsaklis (2003); Munoz de la Pena and Christofides (2008) and the references therein).

In addition to transmitting the data at discrete time instances, another important way of conserving the WSN resources is to select and activate only a subset of the deployed sensor suites at any given time to communicate with the rest of the plant. Under this restriction, the stability and performance characteristics of each unit in the plant become dependent not only on the controller design but also on the selection of the scheduling strategy that, at any time, determines the order in which the sensor suites of the neighboring units transmit their data. Forcing the different subsystems to transmit their data at different times creates opportunities for providing a more targeted correction to the models' estimation errors, such that the models with the largest uncertainties can receive more timely updates than is feasible under the simultaneous transmissions configuration.

Motivated by these considerations, we present in this work an integrated approach for model-based control, state estimation and sensor scheduling in plants with limited state measurements and interconnected processing units that communicate over a resource-constrained WSN. The objective is to find a strategy for establishing and terminating communication between the sensors suites of the WSN and the local control systems in a way that minimizes the rate at which each node in the WSN broadcasts data to the rest of the plant without jeopardizing closed-loop stability. The rest of the paper is organized as follows. Following some preliminaries in Section 2, the networked control and scheduling problem is formulated. Section 3 then presents the quasi-decentralized output feedback control structure and describes its implementation over a WSN with the aid of appropriate local state observers, process models and sensor transmission scheduling. The closed-loop system is then formulated and analyzed in Section 4 where precise conditions for closed-loop stability are provided in terms of the communication rate over the WSN, the sensor scheduling strategy, as well as the accuracy of the models and the choice of controller and observer designs. We show how the stability criteria provide systematic tools that can guide the search for optimal transmission schedules that achieve the biggest savings in WSN resource utilization. Finally, the theoretical results are illustrated in Section 5 using a chemical plant example.

2. PRELIMINARIES

2.1 Plant description

We consider a large-scale distributed plant composed of n interconnected processing units, represented by the following state-space description:

$$\begin{aligned}
\dot{x}_1 &= A_1x_1 + B_1u_1 + \sum_{j=2}^{n} A_{1j}x_j, \qquad y_1 = C_1x_1 \\
\dot{x}_2 &= A_2x_2 + B_2u_2 + \sum_{j=1,j\neq 2}^{n} A_{2j}x_j, \; y_2 = C_2x_2 \\
&\vdots \qquad\qquad\qquad\qquad\qquad\qquad \vdots \\
\dot{x}_n &= A_nx_n + B_nu_n + \sum_{j=1}^{n-1} A_{nj}x_j, \qquad y_n = C_nx_n
\end{aligned} \tag{1}$$

where $x_i := [x_i^{(1)} \; x_i^{(2)} \; \cdots \; x_i^{(p_i)}]^T \in \mathbb{R}^{p_i}$ denotes the vector of process state variables associated with the i-th processing unit, p_i is the number of state variables in the i-th unit, $y_i := [y_i^{(1)} \; y_i^{(2)} \; \cdots \; y_i^{(q_i)}]^T \in \mathbb{R}^{q_i}$ and $u_i := [u_i^{(1)} \; u_i^{(2)} \; \cdots \; u_i^{(r_i)}]^T \in \mathbb{R}^{r_i}$ denote the vector of measured outputs and manipulated inputs associated with the i-th processing unit, respectively, x^T denotes the transpose of a column vector x; A_i, B_i, A_{ij} and C_i are constant matrices. The interconnection term $A_{ij}x_j$, where $i \neq j$, describes how the dynamics of the i-th unit are influenced by the j-th unit in the plant. Note from the summation notation in Eq.1 that each processing unit can in general be connected to all the other units in the plant.

2.2 Problem formulation and solution methodology

Referring to plant of Eq.1, we consider a quasi-decentralized control structure in which each unit in the plant has a local control system with its sensors and actuators connected to the local controller through a dedicated wired communication network. An additional suite of wireless sensors is deployed within each unit to transfer data from the local control system to the plant supervisor as well as to the other distributed control systems in the plant. The various sensor suites form a plant-wide WSN through which the plant units and their controllers communicate. The control objective is to stabilize all the plant units at the zero steady-state while simultaneously: (a) keeping the data dissemination and exchange over the WSN to a minimum, and (b) accounting for the lack of full-state measurements within each unit. To address the resource-constraints problem, we develop in the next section an

integrated model-based quasi-decentralized output feedback control and scheduling strategy that reduces the exchange of information between the plant units without loss of stability. This is accomplished by: (a) designing for each local control system an appropriate state observer that generate estimates of the local state variables from the measured outputs, (b) including models within each control system to estimate the interaction terms when measurements are not available through the WSN, and (c) limiting the number of WSN nodes that, at any given time, transmit their data to update the corresponding target models. The problem is to find an optimal scheduling strategy for establishing and terminating communication between the sensor suites and the target controllers. To illustrate the main ideas, we will consider as an example the configuration where only one wireless sensor suite is allowed to transmit its data to the appropriate units at any given time, while the other nodes remain dormant until the next suite is allowed to transmit its data.

3. QUASI-DECENTRALIZED STATE ESTIMATION AND CONTROL WITH SCHEDULED SENSOR TRANSMISSIONS

3.1 Synthesis of distributed output feedback controllers

Referring to the plant of Eq.1, we begin by synthesizing for each unit an output feedback controller of the form:

$$\begin{aligned} u_i &= K_i\bar{x}_i + \sum_{j=1,j\neq i}^{n} K_{ij}\bar{x}_j \\ \dot{\bar{x}}_i &= (A_i - L_iC_i)\bar{x}_i + \sum_{j=1,j\neq i}^{n} A_{ij}\bar{x}_j + B_iu_i + L_iy_i, \end{aligned} \tag{2}$$

where $\bar{x}_i$ is an estimate of the state of the i-th unit generated by an observer embedded within the local control system of the i-th unit, K_i is the local feedback gain responsible for stabilizing the i-th subsystem in the absence of interconnections, K_{ij} is a gain that compensates for the effect of the j-th neighboring subsystem on the dynamics of the i-th unit, and L_i is the observer gain (chosen such that $A_i - L_iC_i$ is Hurwitz). Note that, in addition to $\bar{x}_i$ which is supplied continuously by the local observer, the implementation of the controller of Eq.2 requires the availability of observer-generated state estimates from the other units in the plant, $\bar{x}_j$, which can be transmitted only through the WSN. A copy of the local observer must therefore be included within the wireless sensor suite of each unit in order to generate the state estimates which are then broadcast to the rest of the plant. This setup is possible given the computational capabilities of wireless sensors. An alternative approach, which avoids having the wireless sensors carry the computational load of the observer, is to have the WSN nodes transmit only the output measurements, but include within each control system an observer of the full plant instead (not just an observer of the local subsystem) which then generates the required state estimates of the full plant state. In addition to the complexity of designing a centralized observer for the entire plant, another difficulty with this approach is that the observer must be designed to have hybrid dynamics since the WSN data are transmitted only at discrete time instances while the local measurements are supplied continuously (or at least more frequently).

It should also be noted that the choice to use a Luenberger observer is made only to illustrate the design and implementation of the quasi-decentralized output feedback control architecture. This choice, however, is not unique and any other explicit observer design can be used instead. The only requirement is that the observer possess an explicit evolution equation that relates the dynamics of the state estimate explicitly to the plant matrices, the output and the observer design parameters. As we will see in the next section, this feature permits the derivation of explicit closed-loop stability conditions that depend in a transparent way on the observer design parameters.

3.2 Design of model-based networked control structure

To conserve battery power in the plant-wide WSN, we initially reduce the rate at which the information (i.e., $\bar{x}_j$) is transferred from the wireless sensor suite of each unit to the target control systems in the neighboring units as much as possible without sacrificing closed-loop stability. To this end, and following the idea presented in (Sun and El-Farra (2008b)), we embed in each unit (both in the local controller and in the wireless sensor suite) a set of dynamic models that provide estimates of the evolution of the states of the neighboring units when communication over the WSN is suspended. The model estimates are used to generate both the local state estimates and the local control action. The state of each model is then reset using the state estimate generated by the observer of the corresponding unit when the wireless sensor suite of the latter is allowed to transmit its data at discrete time instances. In mathematical terms, the local control and update laws for unit i are implemented as follows:

$$\begin{aligned} u_i(t) &= K_i\bar{x}_i(t) + \sum_{j=1,j\neq i}^{n} K_{ij}\widehat{x}_j^i(t),\ t \neq t_k^j,\ i = 1,2,\cdots,n \\ \dot{\bar{x}}_i(t) &= (A_i - L_iC_i)\bar{x}_i(t) + \sum_{j=1,j\neq i}^{n} A_{ij}\widehat{x}_j^i(t) + B_iu_i(t) + L_iy_i(t) \\ \dot{\widehat{x}}_j^i(t) &= \widehat{A}_j\widehat{x}_j^i(t) + \widehat{B}_j\widehat{u}_j^i(t) + \widehat{A}_{ji}\bar{x}_i(t) + \sum_{l=1,l\neq i,l\neq j}^{n} \widehat{A}_{jl}\widehat{x}_l^i(t), t \neq t_k^j \\ \widehat{u}_j^i(t) &= K_j\widehat{x}_j^i(t) + K_{ji}\bar{x}_i(t) + \sum_{l=1,l\neq i,l\neq j}^{n} K_{jl}\widehat{x}_l^i(t),\ \ t \neq t_k^j \\ \widehat{x}_j^i(t_k^j) &= \bar{x}_j(t_k^j),\ \ j = 1,\cdots,n, j \neq i,\ \ k = 0,1,2,\cdots \end{aligned} \tag{3}$$

where $\widehat{x}_j^i$ is the estimate of x_j provided by a model of unit j embedded in unit i; $\widehat{A}_j$, $\widehat{B}_j$ and $\widehat{A}_{jl}$ are constant matrices; t_k^j indicates the k-th transmission time for the j-th sensor suite in the WSN. The fact that $\bar{x}_i$ appears directly in the model of the j-th unit follows from: (1) the structure of the plant and the way the i-th and j-th units are interconnected, and (2) the fact that the observer-generated estimates of x_i are assumed to be available continuously to the local control system of the i-th unit. Note that the models used by the i-th controller to recreate the behavior of the neighboring units do not necessarily match the actual dynamics of those processes, i.e., in general $\widehat{A}_j \neq A_j$, $\widehat{B}_j \neq B_j$, $\widehat{A}_{jl} \neq A_{jl}$.

3.3 Scheduling WSN transmissions and model updates

A key measure of the extent of WSN utilization is the update period for each sensor suite, $h^j := t_{k+1}^j - t_k^j$, which determines the frequency at which the j-th node sends observer estimates to the other units through the network to update the corresponding model states. A larger h implies

larger savings in WSN resource utilization. To simplify the analysis, we consider in what follows only the case when the update period is constant and the same for all the units, so that $t_{k+1}^j - t_k^j := h,\ j = 1, 2, \cdots, n$. To further reduce network utilization, we perform sensor scheduling whereby only one wireless sensor suite is allowed to transmit its observer estimates to the appropriate units at any one time, while the other suites remain dormant until the next suite is allowed to transmit its data (the analysis can be generalized to cases where multiple suites transmit at the same time). The transmission schedule is defined by: (1) the sequence (or order) of transmitting nodes: $\{s_j,\ j = 1, 2, \cdots, n\},\quad s_j \in \mathcal{N} := \{1, 2, \cdots, n\}$, where s_j is a discrete variable that denotes the j-th transmitting entity in the sequence, and (2) the time at which each node in the sequence transmits observer estimates. To characterize the transmission times, we introduce the variable: $\Delta t_j := t_k^{s_{j+1}} - t_k^{s_j},\ \ j = 1, 2, \cdots, n-1$, which is the time interval between the transmissions of two consecutive nodes in the sequence. Fig.1 is a schematic representation of how

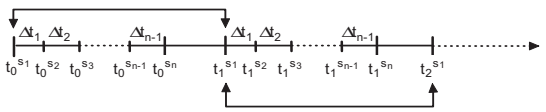


Fig. 1. A schematic showing the time-line for the transmission of each sensor suite in an h-periodic schedule.

sensor scheduling is performed. Note that the schedule is h-periodic in that the same sequence of transmitting nodes is executed repeatedly every h seconds (equivalently, each node transmits its data every h seconds). Note also from the definitions of both h and Δt_j that we always have the constraint $\sum_{j=1}^{n-1} \Delta t_j < h$. Since the update periods for all units are the same, the intervals between the transmission times of two specific units are constant, and within any single execution of the schedule (which lasts less than h seconds), each sensor suite can only transmit its observer estimates through the WSN and update its target models in the local control systems of its neighbors once. This can be represented mathematically by the condition: $s_i \neq s_j$ when $i \neq j$. By manipulating the time intervals Δt_j (i.e., the transmission times) and the order in which the nodes transmit, one can systematically search for the optimal sensor transmission schedule that leads to the largest update period (or smallest communication rate between each sensor suite and its target units).

4. NETWORKED CLOSED-LOOP STABILITY ANALYSIS

4.1 Characterizing the scheduled closed-loop response

In order to derive conditions for closed-loop stability, we need first to express the plant response as a function of the update period and the sensor transmission schedule. To this end, we define the model estimation errors by $e_j^i = \bar{x}_j - \widehat{x}_j^i$, for $j \neq i$, and $e_j^i = 0$, for $j = i$, where e_j^i represents the difference between the state of the observer of unit j (embedded in unit j) and the state of the model of unit j (embedded in unit i). Introducing the augmented vectors: $\mathbf{e}_j := [(e_j^1)^T\ (e_j^2)^T\ \cdots\ (e_j^n)^T]^T$, $\mathbf{e} := [\mathbf{e}_1^T\ \mathbf{e}_2^T\ \cdots\ \mathbf{e}_n^T]^T$, $\mathbf{x} := [x_1^T\ x_2^T\ \cdots\ x_n^T]^T$, $\bar{\mathbf{x}} := [\bar{x}_1^T\ \bar{x}_2^T\ \cdots\ \bar{x}_n^T]^T$, it can be shown that the overall closed-loop plant of Eq.1 and Eq.3 can be formulated as a combined discrete-continuous system of the form:

$$\begin{aligned}
\dot{\mathbf{x}}(t) &= \Lambda_{11}\mathbf{x}(t) + \Lambda_{12}\bar{\mathbf{x}}(t) + \Lambda_{13}\mathbf{e}(t)\\
\dot{\bar{\mathbf{x}}}(t) &= \Lambda_{21}\mathbf{x}(t) + \Lambda_{22}\bar{\mathbf{x}}(t) + \Lambda_{23}\mathbf{e}(t)\\
\dot{\mathbf{e}}(t) &= \Lambda_{31}\mathbf{x}(t) + \Lambda_{32}\bar{\mathbf{x}}(t) + \Lambda_{33}\mathbf{e}(t),\ \ t \neq t_k^j\\
\mathbf{e}_j(t_k^j) &= \mathbf{0},\ \ j = 1, 2, \cdots, n,\ \ k = 0, 1, 2, \cdots,
\end{aligned} \tag{4}$$

where Λ_{ij}'s are constant matrices whose explicit forms are omitted for brevity but can be obtained by substituting Eq.3 into Eq.1. Note that, unlike the case of simultaneous sensor transmissions (where no scheduling takes place) which was investigated in Sun and El-Farra (2008b), not all models within a given unit are updated (and hence not all estimation errors are re-set to zero) at each transmission time. Instead, only the model of the transmitting unit is updated using the observer-generated estimates provided by the wireless sensor suite of that particular unit. Defining the augmented state vector $\xi(t) := [\mathbf{x}^T(t)\ \bar{\mathbf{x}}^T(t)\ \mathbf{e}^T(t)]^T$, the dynamics of the overall closed-loop system can be cast in the following form:

$$\begin{aligned}
\dot{\xi}(t) &= \Lambda_o \xi(t),\ \ t \neq t_k^j\\
\xi(t_k^j) &= \left[\mathbf{x}^T(t_k^j)\ \ \bar{\mathbf{x}}^T(t_k^j)\ \ \mathbf{e}^T(t_k^j)\right]^T,\ k = 0, 1, 2, \cdots\\
\mathbf{e}^T(t_k^j) &= \left[\mathbf{e}_1^T(t_k^j)\ \cdots\ \mathbf{e}_{j-1}^T(t_k^j)\ \ \mathbf{0}\ \ \mathbf{e}_{j+1}^T(t_k^j)\ \cdots\ \mathbf{e}_n^T(t_k^j)\right]^T\\
\Lambda_o &= \begin{bmatrix} \Lambda_{11} & \Lambda_{12} & \Lambda_{13} \\ \Lambda_{21} & \Lambda_{22} & \Lambda_{23} \\ \Lambda_{31} & \Lambda_{32} & \Lambda_{33} \end{bmatrix}
\end{aligned} \tag{5}$$

The following proposition provides an explicit characterization of the scheduled closed-loop response in terms of the update period and the transmission schedule. The proof can be obtained by solving the system of Eq.5 within each sub-interval in Fig.1, and is omitted for brevity.

Proposition 1. Consider the closed-loop system described by Eq.5 with a transmission schedule $\{s_1, s_2, \cdots, s_n\}$ and the initial condition $\xi(t_0^{s_1}) = [\mathbf{x}^T(t_0^{s_1})\ \ \bar{\mathbf{x}}^T(t_0^{s_1})\ \ \mathbf{e}^T(t_0^{s_1})]^T = \xi_0$, with $\mathbf{e}_{s_1}(t_0^{s_1}) = \mathbf{0}$. Then, for $k = 0, 1, 2, \cdots$,

$$\xi(t) = \begin{cases} e^{\Lambda_o(t - t_k^{s_j})}\Gamma_j(\Delta t_j, I_o^{s_j})M_o^k\xi_0,\ t \in [t_k^{s_j},\ t_k^{s_{j+1}}) \\ e^{\Lambda_o(t - t_k^{s_n})}\Gamma_n M_o^k\xi_0,\ t \in [t_k^{s_n},\ t_{k+1}^{s_1}) \end{cases} \tag{6}$$

where $j = 1, 2, \cdots, n-1$, and

$$\Gamma_j = \prod_{j-1-\mu=0}^{j-2} I_o^{s_{\mu+1}} e^{\Lambda_o \Delta t_\mu}, \text{for } j \geq 2, \text{and } \Gamma_j = I,\ \text{for } j = 1 \tag{7}$$

$$M_o = I_o^{s_1} e^{\Lambda_o(h - \sum_{j=1}^{n-1}\Delta t_j)}\Gamma_n \tag{8}$$

$$I_o^{s_j} = \begin{bmatrix} I & O & \cdots & O \\ O & H_1 & \cdots & O \\ \vdots & \vdots & & \vdots \\ O & O & \cdots & H_n \end{bmatrix},\ H_i = \begin{cases} I,\ i \neq s_j \\ O,\ i = s_j \end{cases} \tag{9}$$

for $j = 1, 2, \cdots, n$, $t_{k+1}^{s_j} - t_k^{s_j} = h$ and $\Delta t_j = t_k^{s_{j+1}} - t_k^{s_j},\ j = 1, 2, \cdots, n-1$.

4.2 Characterizing the maximum allowable update period

Having expressed the overall closed-loop response in terms of the update period, the transmission times (which are determined by Δt_j) and the sequence of transmitting nodes (which determines the structure of $I_o^{s_j}$), we are in a position to state the main result of this section. The following theorem provides a necessary and sufficient condition for stability of the scheduled closed-loop plant under the quasi-decentralized networked output feedback control structure. The proof is omitted for brevity.

Theorem 2. Referring to the scheduled closed-loop system of Eq.5 whose solution is given by Eqs.6-9, the zero solution, $\xi = [\mathbf{x}^T\ \bar{\mathbf{x}}^T\ \mathbf{e}^T]^T = [\mathbf{0}\ \mathbf{0}\ \mathbf{0}]^T$, is globally exponentially stable if and only if the eigenvalues of the matrix in Eq.8 are strictly inside the unit circle.

By examining the structure of the test matrix M_o in Eq.8, it can be seen that its eigenvalues depend on the update period h, the closed-loop matrix Λ_o (which in turn depends on the plant-model mismatch as well as the controller and observer gains for all the units), the time intervals between sensor transmissions $\Delta t_1, \Delta t_2, \cdots, \Delta t_{n-1}$, as well as the sensor transmission sequence $\{s_1, s_2, \cdots, s_n\}$. The stability criteria in Theorem 2 can therefore be used to compare different schedules (by varying the transmission sequence as well as the transmission times) to determine the ones that require the least communication rate between the sensors and the target controllers and therefore produce the biggest savings in WSN battery power utilization. For a fixed schedule, the stability criteria can also be used to compare different models, as well as different controllers and state observers in terms of their robustness with respect to communication suspension (i.e., which ones require measurement updates less frequently than others). Note that choosing $\Delta t_1 = \Delta t_2 = \cdots = \Delta t_{n-1} = 0$ reduces the problem to one where all the nodes in the WSN transmit their observer estimates simultaneously. As expected, in this case stability of the networked closed-loop system depends only on Λ_o and h.

5. SIMULATION STUDY: APPLICATION TO CHEMICAL REACTORS WITH RECYCLE

We consider a plant composed of three non-isothermal continuous stirred-tank reactors (CSTRs) in a cascade. The reactant species A is consumed in each reactor by three parallel irreversible exothermic reactions. The output of the third CSTR is passed through a separator that removes the products and recycles unreacted A to the first CSTR. Under standard modeling assumptions, a plant model of the following form can be derived from conservation laws:

$$\begin{aligned}\frac{dT_j}{dt} &= \frac{F_j^0}{V_j}(T_j^0 - T_j) + \frac{F_{j-1}}{V_j}(T_{j-1} - T_j) \\ &\quad + \sum_{i=1}^{3}\frac{(-\Delta H_i)}{\rho c_p}R_i(C_{Aj}, T_j) + \frac{Q_j}{\rho c_p V_j} \\ \frac{dC_{Aj}}{dt} &= \frac{F_j^0}{V_j}(C_{Aj}^0 - C_{Aj}) + \frac{F_{j-1}}{V_j}(C_{A(j-1)} - C_{Aj}) \\ &\quad - \sum_{i=1}^{3}R_i(C_{Aj}, T_j),\ j = 1, 2, 3\end{aligned}$$

where T_j, C_{Aj}, Q_j, and V_j denote the temperature, the reactant concentration, the rate of heat input, and the volume of the j-th reactor, respectively, $R_i(C_{Aj}, T_j) = k_{i0}\exp\left(\frac{-E_i}{RT_j}\right)C_{Aj}$ is the rate of the i-th reaction, F_j^0 denotes the flow rate of a fresh feed stream associated with the j-th reactor, F_j is the flow rate of the outlet stream of the j-th reactor, with $F_0 = F_r, T_0 = T_3, C_{A0} = C_{A3}$ denoting the flow rate, temperature and reactant concentration of the recycle stream, ΔH_i, k_i, E_i, $i = 1, 2, 3$, denote the enthalpies, pre-exponential constants and activation energies of the three reactions, respectively, c_p and ρ denote the heat capacity and density of fluid in the reactor. Using typical values for the process parameters (see Sun and El-Farra (2008a)), the plant with $Q_j = 0$, $C_{Aj}^0 = C_{Aj}^{0s}$ and a recycle ratio of $r = 0.5$, has three steady-states (two locally asymptotically stable and one unstable). The control objective is to stabilize the plant at the (open-loop) unstable steady-state by manipulating Q_j and C_{Aj}^0, $j = 1, 2, 3$. Only the temperatures of the three reactors are assumed to be available as measurements. A plant-wide WSN composed of 3 wireless sensor suites is deployed. Each sensor suite collects estimates of the local process state variables provided by a state observer embedded within the unit and broadcasts it to the rest of the plant. It is desired to stabilize the plant with minimal data exchange over the WSN to conserve battery power for the wireless devices.

Linearizing the plant around the unstable steady-state yields a system of the form of Eq.1 to which the networked output feedback control and scheduling architecture described in the previous sections is applied. The synthesis details are omitted due to space limitations. In the remainder of this section, we will investigate the interplay between the communication rate and the sensor transmission schedule, and its impact on closed-loop stability. Since closed-loop stability requires all eigenvalues of M_o to lie within the unit circle, it is sufficient to consider only the maximum eigenvalue magnitude, denoted by $\lambda_{\max}(M_o)$.

Table 1. Sensor transmission schedules

Schedule	$s_1, s_2, s_3, s_1, s_2, s_3, \cdots$
1	$1, 2, 3, 1, 2, 3, \cdots$
2	$1, 3, 2, 1, 3, 2, \cdots$
3	$2, 1, 3, 2, 1, 3, \cdots$
4	$2, 3, 1, 2, 3, 1, \cdots$
5	$3, 1, 2, 3, 1, 2, \cdots$
6	$3, 2, 1, 3, 2, 1, \cdots$

We consider first the case when $\Delta t_1 = \Delta t_2 = \Delta t$. Fig.2(a) is a contour plot showing the dependence of $\lambda_{\max}(M_o)$ on both the interval between transmissions, Δt, and the update period, h, under the six possible sensor transmission schedules listed in Table 1 when imperfect models are embedded in the local control systems (each model has 10% parametric uncertainty in the heat of reaction). For each schedule, the area enclosed by the unit contour line is the stability region of the plant. It can be seen that, for sufficiently small Δt (below 0.03 hr), the maximum allowable update periods obtained under sequences 2 and 6 are larger than the one obtained when no scheduling takes place (i.e., with $\Delta t = 0$). As Δt is increased, however, the trend is reversed, indicating that the benefits of scheduling can be limited by a poor choice of the transmission times. For sequences 3 and 5, scheduling yields larger update periods (compared with the concurrent transmission configuration) only when the transmission times are chosen such that $\Delta t > 0.04$ hr. In general, allowing the different sensor suites to transmit their data and update their target models at different times (rather than simultaneously) can help provide a more targeted and timely (though only partial) correction to model estimation errors which in turn helps reduce the rate at which each node in the WSN must transmit its data. These predictions are further confirmed by the closed-loop state profile shown in Fig.2(b), which shows that the linearized plant is stable under sequence 6 but unstable under sequence 2, when $\Delta t = 0.02$ hr and

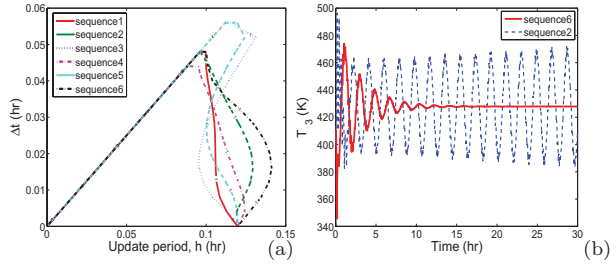


Fig. 2. (a) Dependence of $\lambda_{\max}(M_o)$ on Δt and h for different sensor transmission sequences under a model-based control scheme. (b) Closed-loop temperature profile for CSTR3 under the model-based quasi-decentralized output feedback control strategy using two different sensor transmission schedules with the same update period.

$h = 0.13$ hr (for brevity, only the temperature profile for CSTR 3 is shown; the state and input profiles for the other reactors exhibit similar behavior).

We consider next the more general case where $\Delta t_1 \neq \Delta t_2$. Fig.3(a) is a contour plot showing the dependence of $\lambda_{\max}(M_o)$ on Δt_1 and h for different values of Δt_2, when the WSN nodes transmit according to sequence 2 and an uncertain model is used (nominal value of the heat of reaction is 10% higher than the actual value). It can be seen that a larger update period (and hence larger reduction in WSN utilization) can be obtained by carefully choosing the transmission times for the sensor suites of different units than in the case when $\Delta t_1 = \Delta t_2$. For example, consider the case when $\Delta t_2 = 0.02$ hr and $h = 0.13$ hr. This point lies outside the stability region of schedule 2 when $\Delta t_2 = \Delta t_1 = 0.02$ hr (see Fig.2(a)). If we choose $\Delta t_1 = 0.08$ hr, however, the same update period becomes stabilizing under schedule 2 (the point now lies inside the stability region). These observations are further confirmed by the temperature profiles in Fig.3(b).

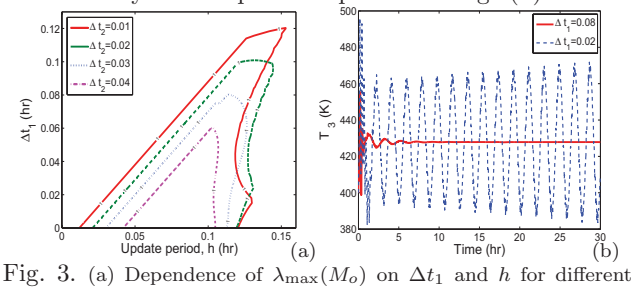


Fig. 3. (a) Dependence of $\lambda_{\max}(M_o)$ on Δt_1 and h for different values of Δt_2 under schedule 2 with a fixed model, and (b) Closed-loop temperature profile for CSTR 3 when $\Delta t_2 = 0.02$ hr and $h = 0.13$ hr for two different values of Δt_1.

REFERENCES

Antelo, L.T., Otero-Muras, I., Banga, J.R., and Alonso, A.A. (2007). A systematic approach to plant-wide control based on thermodynamics. *Comp. & Chem. Eng.*, 31, 677–691.

Baldea, M., Daoutidis, P., and Kumar, A. (2006). Dynamics and control of integrated networks with purge streams. *AIChE J.*, 52, 1460–1472.

Camponogara, E., Jia, D., Krogh, B., and Talukdar, S. (2002). Distributed model predictive control. *IEEE Contr. Syst. Mag.*, 22, 44–52.

Christofides, P.D., Davis, J.F., El-Farra, N.H., Harris, K., Gibson, J., and Clark, D. (2007). Smart plant operations: Vision, progress and challenges. *AIChE J.*, 53, 2734–2741.

Cui, H. and Jacobsen, E. (2002). Performance limitations in decentralized control. *J. Proc. Contr.*, 12, 485–494.

Goodwin, G.C., Salgado, M.E., and Silva, E.I. (2005). Time-domain performance limitations arising from decentralized architectures and their relationship to the RGA. *International Journal of Control*, 78, 1045–1062.

Hangos, K., Alonso, A., Perkins, J., and Ydstie, B. (1999). Thermodynamic approach to the structural stability of process plants. *AIChE J.*, 45, 802–816.

Huang, X. and Huang, B. (2004). Multiloop decentralized PID control based on covariance control criteria: an LMI approach. *ISA Transactions*, 43, 49–59.

Kariwala, V. (2007). Fundamental limitation on achievable decentralized performance. *Automatica*, 43, 1849–1854.

Katebi, M.R. and Johnson, M.A. (1997). Predictive control design for large-scale systems. *Automatica*, 33, 421–425.

Kumar, P.R. (2001). New technological vistas for systems and control: The example of wireless networks. *IEEE Contr. Syst. Mag.*, 21, 24–37.

Lunze, J. (1992). *Feedback Control of Large Scale Systems.* Prentice-Hall, U.K.

Montestruque, L.A. and Antsaklis, P.J. (2003). On the model-based control of networked systems. *Automatica*, 39, 1837–1843.

Munoz de la Pena, D. and Christofides, P.D. (2008). Lyapunov-based model predictive control of nonlinear systems subject to data losses. *IEEE Trans. Automat. Contr.*, 53, 2067–2089.

Siljak, D.D. (1991). *Decentralized Control of Complex Systems.* Academic Press, London.

Skogestad, S. (2004). Control structure design for complete chemical plants. *Comp. & Chem. Eng.*, 28, 219–234.

Song, J., Mok, A.K., Chen, D., and Nixon, M. (2006). Challenges of wireless control in process industry. In *Workshop on Research Directions for Security and Networking in Critical Real-Time and Embedded Systems.* San Jose, CA.

Sourlas, D.D. and Manousiouthakis, V. (1995). Best achievable decentralized performance. *IEEE Trans. Automat. Contr.*, 40, 1858–1871.

Sun, Y. and El-Farra, N.H. (2008a). Quasi-decentralized model-based networked control of process systems. *Comp. & Chem. Eng.*, 32, 2016–2029.

Sun, Y. and El-Farra, N.H. (2008b). Quasi-decentralized state estimation and control of process systems over communication networks. In *Proceedings of the 47th IEEE Conference on Decision and Control*, 5468–5475. Cancun, Mexico.

Tetiker, M.D., Artel, A., Teymour, F., and Cinar, A. (2008). Control of grade transitions in distributed chemical reactor networks: An agent-based approach. *Comp. & Chem. Eng.*, 32, 1984–1994.

Venkat, A.N., Rawlings, J.B., and Wright, S.J. (2005). Stability and optimality of distributed model predictive control. In *Proceedings of the 44th IEEE Conference on Decision and Control*, 6680–6685. Seville, Spain.

Walsh, G. and Ye, H. (2001). Scheduling of networked control systems. *IEEE Contr. Syst. Mag.*, 21, 57–65.

Ydstie, B.E. (2002). New vistas for process control: Integrating physics and communication networks. *AIChE J.*, 48, 422–426.

Bidirectional Branch and Bound Method for Selecting Controlled Variables

Vinay Kariwala * **Yi Cao** **

* *Division of Chemical & Biomolecular Engineering, Nanyang Technological University, Singapore 637459 (e-mail: vinay@ntu.edu.sg)*
** *School of Engineering, Cranfield University, Bedford, UK (e-mail: y.cao@cranfield.ac.uk)*

Abstract: Controlled variable (CV) selection from available measurements through exhaustive search is computationally forbidding for large-scale problems. We have recently proposed novel bidirectional branch and bound (B^3) approaches for CV selection using the minimum singular value (MSV) rule and the local worst-case loss criterion in the framework of self-optimizing control. However, the MSV rule is approximate and worst-case scenario may not occur frequently in practice. In this work, the B^3 approach is extended to CV selection based on the recently developed local average loss metric, which represents the expected loss incurred over the long-term operation of the plant. Lower bounds on local average loss and fast pruning algorithms are derived for the efficient B^3 algorithm. Numerical tests and binary distillation column case study are used to demonstrate the computational efficiency of the proposed method.

Keywords: Branch and bound, Combinatorial optimization, Controlled variable, Self-optimizing control.

1. INTRODUCTION

The selection of controlled variables (CVs) from available measurements is an important task during the design of control systems for complex processes. Traditionally, CVs have been selected based on intuition and process knowledge. To systematically select CVs, Skogestad (2000) introduced the concept of self-optimizing control. In this approach, CVs are selected such that in presence of disturbances, the loss incurred in implementing the operational policy by holding the selected CVs at constant setpoints is minimal, as compared to the use of an online optimizer.

The choice of CVs based on the general non-linear formulation of self-optimizing control requires solving large-dimensional non-convex optimization problems (Skogestad, 2000). To quickly pre-screen alternatives, local methods have been proposed including the minimum singular value (MSV) rule (Skogestad and Postlethwaite, 1996) and exact local methods with worst-case (Halvorsen et al., 2003) and average loss minimization (Kariwala et al., 2008). Though the local methods simplify loss evaluation for a single alternative, every feasible alternative still needs to be evaluated to find the optimal solution. As the number of alternatives grows rapidly with process dimensions, such an exhaustive search is computationally intractable for large-scale processes. Thus, an efficient method is needed to find a subset of available measurements, which can be used as CVs (Problem 1).

Instead of selecting CVs as a subset of available measurements, it is possible to obtain lower losses using combinations of available measurements as CVs (Halvorsen et al., 2003). Recently, explicit solutions to the problem of finding locally optimal measurement combinations have been proposed (Kariwala, 2007; Kariwala et al., 2008; Alstad et al., 2009). It is possible, however, that the use of combinations of a few measurements as CVs may provide similar loss as the case where combinations of all available measurements are used (Kariwala, 2007; Kariwala et al., 2008; Alstad et al., 2009). Though the former approach results in control structures with lower complexity, it gives rise to another combinatorial optimization problem involving the identification of the set of measurements, whose combinations can be used as CVs (Problem 2).

Both Problems 1 and 2 can be seen as subset selection problems, for which only exhaustive search and branch and bound (BAB) method guarantee globally optimal solution. For minimization problems, a BAB approach divides the problem into several sub-problems (nodes) and calculates a lower bound of the selection criterion over all possible solutions of a node. If the lower bound is greater than an upper bound of the optimal solution, then the corresponding node is pruned (eliminated without further evaluation). In this way, the BAB method gains its efficiency in comparison with exhaustive search. The traditional BAB methods for subset selection use downwards approach, where pruning is performed on nodes with gradually decreasing subset size (Narendra and Fukunaga, 1977). Recently, a novel bidirectional BAB (B^3) approach (Cao and Kariwala, 2008) has been proposed for CV selection, where non-optimal nodes are pruned in downwards as well as upwards (gradually increasing subset size) directions simultaneously, which significantly reduces the solution time.

The bidirectional BAB (B^3) approach has been applied to solve Problem 1 with MSV rule (Cao and Kariwala, 2008) and local worst-case loss (Kariwala and Cao, 2009)

as selection criteria. A partially bidirectional BAB (PB^3) method has also been proposed to solve Problem 2 through minimization of local worst-case loss (Kariwala and Cao, 2009). The MSV rule, however, is approximate and can lead to non-optimal set of CVs (Hori and Skogestad, 2008). Selection of CVs based on local worst-case loss minimization can also be conservative, as the worst-case may not occur frequently in practice (Kariwala et al., 2008). Thus, CV selection through minimization of local average loss, which represents the expected loss incurred over the long-term operation of the plant, can be deemed as most reliable.

In this paper, lower bounds on local average loss and fast pruning algorithms are derived to develop an efficient B^3 method for CV selection using the exact local method with average loss minimization. A PB^3 method is also developed to find a subset of available measurements, whose combinations can be used as CVs to minimize local average loss. Numerical tests and binary distillation column case study are used to demonstrate the computational efficiency of the proposed method.

2. BAB METHODS FOR SUBSET SELECTION

Let $X_m = \{x_1, x_2, \cdots, x_m\}$ be an m-element set. The subset selection problem with selection criterion T involves finding an n-element subset $X_n \subset X_m$ such that

$$T(X_n^*) = \min_{X_n \subset X_m} T(X_n) \tag{1}$$

For a subset selection problem, the total number of candidates grows very quickly as m and n increase, which renders exhaustive search unviable. BAB approach can find the globally optimal subset without exhaustive search.

2.1 Unidirectional BAB approaches

Downwards. BAB search is traditionally conducted downwards (gradually decreasing subset size). A downwards solution tree for selecting 2 out of 6 elements is shown in Figure 1(a), where the root node is the same as X_m. Other nodes represent subsets obtained by eliminating one element from their parent sets. Labels at nodes denote the elements discarded there. To describe the pruning principle, let B be an upper bound of the globally optimal criterion, *i.e.* $B \geq T(X_n^*)$ and $\underline{T}_n(X_s)$ be a downwards lower bound over all n-element subsets of X_s, *i.e.* $\underline{T}_n(X_s) \leq T(X_n)\ \forall X_n \subseteq X_s$. Then,

$$T(X_n) > T(X_n^*)\, \forall X_n \subseteq X_s, \text{if } \underline{T}_n(X_s) > B \tag{2}$$

Hence, any n-element subset of X_s cannot be optimal and can be pruned without further evaluation, if $\underline{T}_n(X_s) > B$.

Upwards. Subset selection can also be performed upwards (gradually increasing subset size). An upwards solution tree for selecting 2 out of 6 elements is shown in Figure 1(b), where the root node is an empty set. Other nodes represent supersets obtained by adding one element to their parent sets. Labels at nodes denote the elements added there. To introduce the pruning principle, let the upwards lower bound of the selection criterion be defined as $\underline{T}_n(X_t) \leq T(X_n)\ \forall X_n \supseteq X_t$. Then,

$$T(X_n) > T(X_n^*)\, \forall X_n \supseteq X_t, \text{if } \underline{T}_n(X_t) > B \tag{3}$$

As downwards BAB, if $\underline{T}_n(X_t) > B$, any n-element superset of X_t cannot be optimal and hence can be pruned without further evaluation.

2.2 Bidirectional BAB approach

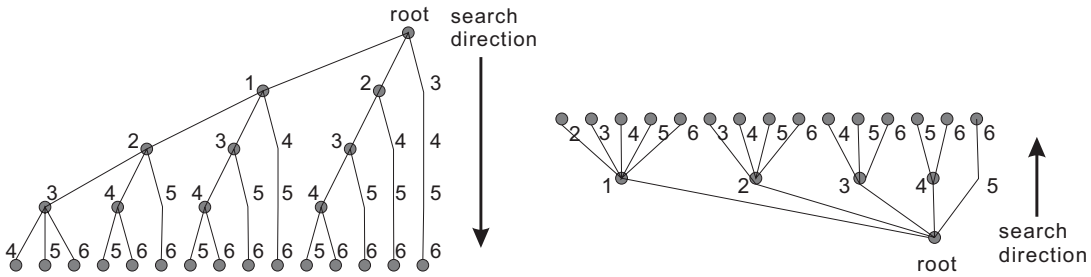
The upwards and downwards BAB approaches can be combined to form a more efficient bidirectional BAB (B^3) approach. This approach is applicable to any subset selection problem, for which both upwards and downwards lower bounds on the selection criterion are available (Cao and Kariwala, 2008).

Bidirectional pruning. In a B^3 approach, the whole subset selection problem is divided into several subproblems. A sub-problem is represented as the 2-tuple $\mathcal{S} = (F_f, C_c)$, where F_f is an f-element fixed set and C_c is a c-element candidate set. Here, $f \leq n$ and $n \leq f + c \leq m$. The elements of F_f are included in all n-element subsets that can be obtained by solving $\mathcal{S}$, while elements of C_c can be freely chosen to append F_f. In terms of fixed and candidate sets, downwards and upwards pruning can be performed if $\underline{T}_n(F_f \cup C_c) > B$ and $\underline{T}_n(F_f) > B$, respectively. In B^3 approach, these pruning conditions are used together (bidirectional pruning), where the subproblem $\mathcal{S}$ is pruned, if either downwards or upwards pruning condition is met.

The use of bidirectional pruning significantly improves the efficiency as non-optimal subproblems can be pruned at an early stage of the search. Further gain in efficiency is achieved by carrying out pruning on the sub-problems of $\mathcal{S}$, instead of on $\mathcal{S}$ directly. For $x_i \in C_c$, upward pruning is conducted by discarding x_i from C_c, if $\underline{T}_n(F_f \cup x_i) > B$. Similarly, if $\underline{T}(F_f \cup (C_c \backslash x_i)) > B$, then downward pruning is performed by moving x_i from C_c to F_f. Here, an advantage of performing pruning on sub-problems is that the bounds $\underline{T}_n(F_f \cup x_i)$ and $\underline{T}_n(F_f \cup (C_c \backslash x_i))$ can be computed from $\underline{T}_n(F_f)$ and $\underline{T}_n(F_f \cup C_c)$, respectively, for all $x_i \in C_c$ together, resulting in computational efficiency.

Bidirectional branching. In downwards and upwards BAB methods, branching is performed by removing elements from C_c and moving elements from C_c to F_f, respectively. These two branching approaches can be combined into an efficient bidirectional approach by selecting a decision element and deciding upon whether the decision element be eliminated from C_c or moved to F_f. In the B^3 algorithm, the decision element is selected as the one with the largest upwards or downwards upper bound for upward or downward search (best-first search), respectively.

The branching direction (upwards or downwards) is selected by comparing the number of terminal nodes (n-element subsets) of the resulting subproblems with alternate approaches such that the simpler branch is evaluated first, whilst the other branch is kept for possible pruning in future. For downwards branching, removing an element from C_c results in a subproblem with $\mathcal{C}_{c-1}^{n-f}$ terminal nodes, whilst for upwards branching, moving an element from C_c to F_f gives a subproblem with $\mathcal{C}_{c-1}^{n-f-1}$ terminal nodes. Therefore, if $2(n-f) > c$, downwards branching is performed, otherwise upwards branching is selected.



(a) downward search (b) upward search

Fig. 1. Solution trees for selecting 2 out of 6 elements.

3. SELF-OPTIMIZING CONTROL

To present the local method for self-optimizing control, consider that the economics of the plant is characterized by the scalar objective functional $J(\mathbf{u}, \mathbf{d})$, where $\mathbf{u} \in \mathbb{R}^{n_u}$ and $\mathbf{d} \in \mathbb{R}^{n_d}$ denote the degrees of freedom or inputs and disturbances, respectively. The linearized model of the process around the nominally optimal operating point is

$$\mathbf{y} = \mathbf{G}^y \, \mathbf{u} + \mathbf{G}^y_d \, \mathbf{W}_d \, \mathbf{d} + \mathbf{W}_e \, \mathbf{e} \tag{4}$$

where $\mathbf{y} \in \mathbb{R}^{n_y}$ denotes the process measurements and $\mathbf{e} \in \mathbb{R}^{n_y}$ represents the implementation error including measurement and control errors. Here, the diagonal matrices $\mathbf{W}_d$ and $\mathbf{W}_e$ contain the magnitudes of expected disturbances and implementation errors associated with the individual measurements, respectively. The CVs $\mathbf{c} \in \mathbb{R}^{n_u}$ are given as

$$\mathbf{c} = \mathbf{H} \, \mathbf{y} = \mathbf{G} \, \mathbf{u} + \mathbf{G}_d \, \mathbf{W}_d \, \mathbf{d} + \mathbf{H} \, \mathbf{W}_e \, \mathbf{e} \tag{5}$$

where $\mathbf{G}_d = \mathbf{H} \, \mathbf{G}^y_d$ and $\mathbf{G} = \mathbf{H} \, \mathbf{G}^y \in \mathbb{R}^{n_u \times n_u}$ is invertible, a necessary condition for integral control.

When $\mathbf{d}$ and $\mathbf{e}$ are constrained to satisfy

$$\left\| \begin{bmatrix} \mathbf{d}^T & \mathbf{e}^T \end{bmatrix} \right\|_2^T \leq 1 \tag{6}$$

Kariwala et al. (2008) have shown that the average loss over the set (6) is given as

$$L_{\text{average}}(\mathbf{H}) = \frac{1}{6(n_y + n_d)} \left\| (\mathbf{H}\tilde{\mathbf{G}})^{-1} \mathbf{H} \mathbf{Y} \right\|_F^2 \tag{7}$$

where $\tilde{\mathbf{G}} = \mathbf{G}^y \mathbf{J}_{uu}^{-1/2}$ and

$$\mathbf{Y} = \begin{bmatrix} (\mathbf{G}^y \, \mathbf{J}_{uu}^{-1} \, \mathbf{J}_{ud} - \mathbf{G}^y_d) \, \mathbf{W}_d & \mathbf{W}_e \end{bmatrix} \tag{8}$$

When individual measurements are selected as CVs, the elements of $\mathbf{H}$ are restricted to be 0 or 1 and $\mathbf{H}\mathbf{H}^T = \mathbf{I}$. Using index notation, this problem can be stated as

$$\min_{X_{n_u} \subset X_{n_y}} L_1(X_{n_u}) = \left\| \tilde{\mathbf{G}}^{-1}_{X_{n_u}} \mathbf{Y}_{X_{n_u}} \right\|_F^2 \tag{9}$$

Note that the scalar constant $1/(6(n_y+n_d))$ is neglected in (9), as it does not depend on the selected CVs. Instead of 2-norm, as used in (6), if a different norm is used to define the allowable set of $\mathbf{d}$ and $\mathbf{e}$, the resulting expressions for average losses only differ by scalar constants (Kariwala et al., 2008). Thus, the formulation of optimization problem in (9) is independent of the norm used to define the allowable set of $\mathbf{d}$ and $\mathbf{e}$.

Instead of using individual measurements, it is possible to use combinations of measurements as CVs. In this case, the integer constraint on $\mathbf{H} \in \mathbb{R}^{n_u \times n_y}$ is relaxed, but the condition $\text{rank}(\mathbf{H}) = n_u$ is still imposed to ensure invertibility of $\mathbf{H}\,\mathbf{G}^y$. The minimal average loss over the set (6) using measurements combinations as CVs is given as (Kariwala et al., 2008)

$$\min_{\mathbf{H}} L_{\text{average}} = \frac{1}{6\,(n_y + n_d)} \sum_{i=1}^{n_u} \lambda_i^{-1} \left(\tilde{\mathbf{G}}^T \, (\mathbf{Y}\,\mathbf{Y}^T)^{-1} \, \tilde{\mathbf{G}} \right) \tag{10}$$

Equation (10) can be used to calculate the minimum loss provided by the optimal combination of a given set of measurements. However, the use of all measurements is often unnecessary and similar losses may be obtained by combining only a few of the available measurements. Then, the combinatorial optimization problem involves finding the set of n among n_y measurements ($n_u \leq n \leq n_y$) that can provide the minimal loss for specified n. In index notation, the n measurements are selected by minimizing

$$\min_{X_n \subset X_{n_y}} L_2(X_n) = \sum_{i=1}^{n_u} \lambda_i^{-1} \left(\tilde{\mathbf{G}}^T_{X_n} (\mathbf{Y}_{X_n} \mathbf{Y}^T_{X_n})^{-1} \tilde{\mathbf{G}}_{X_n} \right) \tag{11}$$

where the scalar constant has been omitted as (9).

4. BAB METHOD FOR CV SELECTION

As shown in Section 3, the selection of CVs using exact local method can be seen as subset selection problems. In this section, the BAB methods for solving these problems are presented. For simplicity of notation, we define the $p \times p$ matrix $\mathbf{M}(X_p)$ and the $n_u \times n_u$ matrix $\mathbf{N}(X_p)$ as

$$\mathbf{M}(X_p) = \mathbf{R}^{-T} \tilde{\mathbf{G}}_{X_p} \tilde{\mathbf{G}}^T_{X_p} \mathbf{R}^{-1} \tag{12}$$

$$\mathbf{N}(X_p) = \tilde{\mathbf{G}}^T_{X_p} (\mathbf{Y}_{X_p} \mathbf{Y}^T_{X_p})^{-1} \tilde{\mathbf{G}}_{X_p} \tag{13}$$

where $\mathbf{R}$ is the Cholesky factor of $\mathbf{Y}_{X_p} \mathbf{Y}^T_{X_p}$.

4.1 Lower bounds

Individual measurements. L_1 in (9) requires inversion of $\mathbf{G}_{X_{n_u}}$ and thus $L_1(X_p)$ is well-defined only when $\mathbf{G}_{X_p}$ is a square matrix, *i.e.* $p = n_u$. On the other hand, BAB methods require evaluation of loss, when the number of selected measurements differs from n_u. Motivated by this drawback, two alternate representations of L_1 are derived as follows:

$$L_1(X_p) = \sum_{i=1}^{r} \lambda_i^{-1}\left(\mathbf{N}(X_p)\right) = \sum_{i=1}^{r} \lambda_i^{-1}\left(\mathbf{M}(X_p)\right) \quad (14)$$

where $r = \text{rank}(\tilde{\mathbf{G}}_{X_p})$. It is clear that for $r = p = n_u$, (14) is equivalent to (9). However, (14) generally holds for any number of measurements since $\mathbf{Y}_{X_p}\mathbf{Y}_{X_p}^T$ is invertible under the reasonable assumption that every measurement has a non-zero implementation error. Using the generalized expression for L_1 and interlacing properties of eigenvalues (Horn and Johnson, 1985), the downwards and upwards lower bounds required for the application of B^3 algorithm are derived as follows.

Proposition 1. (Lower bounds for L_1). Consider a node $\mathcal{S} = (F_f, C_c)$. For L_1 defined in (14),

$$L_1(F_f) \leq \min_{X_{n_u} \supset F_f} L_1(X_{n_u});\ f < n_u \quad (15)$$

$$L_1(F_f \cup C_c) \leq \min_{X_{n_u} \subset (F_f \cup C_c)} L_1(X_{n_u});\ f + c > n_u \quad (16)$$

To illustrate the implications of Proposition 1, let B represent the best available upper bound on $L_1(X^*_{n_u})$. Then (15) implies that, if $L_1(F_f) > B$, the optimal solution cannot be a superset of F_f and hence all supersets of F_f need not be evaluated. Similarly, if $L_1(F_f \cup C_c) > B$, (16) implies that the optimal solution cannot be a subset of $F_f \cup C_c$ and hence all subsets of $F_f \cup C_c$ need not be evaluated. Thus, upwards and downwards pruning can be conduced using (15) and (16) and the optimal solution can be found without complete enumeration.

Measurements combinations. The expression for L_2 in (11) is the same as the expression for L_1 in (14). Thus, similar to Proposition 1, it can be shown that

$$L_2(F_f \cup C_c) \leq \min_{X_n \subset (F_f \cup C_c)} L_2(X_n); \quad f + c > n \quad (17)$$

For selecting measurements, whose combinations can be used as CVs, the result in (17) is useful for downwards pruning. Equation (16), however, also implies that when $n_u \leq f < n$, $L_2(F_f)$ decreases as the subset size increases. Thus, unlike L_1, the expression for L_2 cannot be directly used for upwards pruning. In the following proposition, a lower bound on L_2 is derived, which can instead be used for upwards pruning, whenever $n - n_u < f < n$.

Proposition 2. (Upwards lower bound for L_2). For the node $\mathcal{S} = (F_f, C_c)$, let

$$\underline{L}_2(F_f) = \sum_{i=1}^{f+n_u-n} \lambda_i^{-1}\left(\mathbf{N}(F_f)\right) \quad (18)$$

where $f > n - n_u$. Then, $\underline{L}_2(F_f)$ represents a lower bound on the loss corresponding to combinations of any n measurements obtained by appending indices to F_f, *i.e.*

$$\underline{L}_2(F_f) \leq \min_{\substack{X_n \supset F_f \\ X_n \subset (F_f \cup C_c)}} L_2(X_n) \quad (19)$$

Proposition 2 implies that the lower bound of L_2 defined in (18) can be used for upwards pruning. In this case, upwards pruning can only be applied when the size of fixed set of the node under consideration is greater than $n - n_u$. Thus, the BAB algorithm based on $\underline{L}_2$ in (18) is referred to as partially bidirectional BAB (PB^3) algorithm. Development of fully bidirectional BAB algorithm for selection of measurement combination as CVs is an open problem.

4.2 Fast pruning and branching

Propositions 1 and 2 can be used to prune the non-optimal nodes quickly. Thus, the optimal solution can be found with evaluation of fewer nodes, but the solution time can still be large, as direct evaluation of L_1 in (14) and L_2 in (11) requires eigenvalue decomposition, which is computationally expensive.

Individual measurements. When $f < n_u$, $\mathbf{M}(F_f)$ in (12) is invertible. Similarly when $s = f + c > n_u$, $\mathbf{N}(S_s)$ in (13) for $S_s = F_f \cup C_c$ is invertible. Thus,

$$L_1(F_f) = \sum_{i=1}^{r} \lambda_i^{-1}(\mathbf{M}(F_f)) = \text{trace}(\mathbf{M}^{-1}(F_f)) \quad (20)$$

$$L_1(S_s) = \sum_{i=1}^{r} \lambda_i^{-1}(\mathbf{N}(S_s)) = \text{trace}(\mathbf{N}^{-1}(S_s)) \quad (21)$$

The use of (20) and (21) for evaluation of lower bounds on L_1 avoids computation of eigenvalues. The next two propositions relate the bounds of a node with the bounds of sub-nodes allowing pruning on sub-nodes directly and thus improving efficiency of the B^3 algorithm further.

Proposition 3. (Upwards pruning for L_1). Consider a node $\mathcal{S} = (F_f, C_c)$ and index $i \in C_c$. Then

$$L_1(F_f \cup i) = L_1(F_f) + \frac{\|\mathbf{z}_i^T\mathbf{Y}_{F_f} - \mathbf{Y}_i\|_2^2}{\eta_i} \quad (22)$$

where $\mathbf{z}_i = (\tilde{\mathbf{G}}_{F_f}\tilde{\mathbf{G}}_{F_f}^T)^{-1}\tilde{\mathbf{G}}_{F_f}\tilde{\mathbf{G}}_i^T$ and $\eta_i = \tilde{\mathbf{G}}_i(\mathbf{I} - \mathbf{G}_{F_f}^T(\tilde{\mathbf{G}}_{F_f}\tilde{\mathbf{G}}_{F_f}^T)^{-1}\tilde{\mathbf{G}}_{F_f})\tilde{\mathbf{G}}_i^T$.

Proposition 4. (Downward pruning for L_1). For a node $\mathcal{S} = (F_f, C_c)$, let $S_s = F_f \cup C_c$, where $s = f + c$. For $i \in C_c$,

$$L_1(S_s \setminus i) = L_1(S_s) + \frac{\|\mathbf{x}_i\mathbf{N}^{-1}(S_s)\|_2^2}{\zeta_i - \mathbf{x}_i\mathbf{N}^{-1}(S_s)\mathbf{x}_i^T} \quad (23)$$

where $\mathbf{x}_i = \mathbf{Y}_i\mathbf{Y}_{S_s\setminus i}^T(\mathbf{Y}_{S_s\setminus i}\mathbf{Y}_{S_s\setminus i}^T)^{-1}\mathbf{G}_{S_s\setminus i} - \mathbf{G}_i^T$ and $\zeta_i = \mathbf{Y}_i(\mathbf{I} - \mathbf{Y}_{S_s\setminus i}^T(\mathbf{Y}_{S_s\setminus i}\mathbf{Y}_{S_s\setminus i}^T)^{-1}\mathbf{Y}_{S_s\setminus i})\mathbf{Y}_i^T$.

In comparison with the direct calculation of L_1, the use of (22) and (23) is computationally less demanding. This happens as in (22), the inverse $(\tilde{\mathbf{G}}_{F_f}\tilde{\mathbf{G}}_{F_f}^T)^{-1}$ needs to be evaluated only once for all c sub-nodes, whilst in (23), two inverses $(\tilde{\mathbf{Y}}_{S_s\setminus i}\tilde{\mathbf{Y}}_{S_s\setminus i}^T)^{-1}$ and $N^{-1}(S_s)$ are evaluated only once for all c sub-nodes.

Measurements combinations. As the downwards pruning criteria for minimization of L_1 and L_2 are the same, Proposition 4 can be used for fast downwards pruning for selection of a subset of measurements, whose combinations can be used as CVs. The fast upwards pruning criteria for minimization of L_2 is presented in the next proposition.

Proposition 5. (Upwards pruning for L_2). Consider a node $\mathcal{S} = (F_f, C_c)$ and index $i \in C_c$. Then

$$\underline{L}_2(F_f \cup i) \geq \sum_{j=1}^{f+n_u-n+1} \frac{1}{\lambda_j(\mathbf{N}(F_f)) + t_j} \tag{24}$$

where $t = [t_1 \ \cdots \ t_{f+n_u-n+1}]^T$ is determined by solving the following linear equations:

$$t_j - t_{j+1} = \lambda_{j+1} - \lambda_j, \ j = 1, 2, \cdots, f + n_u - n \tag{25}$$

$$\sum_{j=1}^{f+n_u-n+1} t_j = \|\mathbf{s}_i\|_2^2/\beta_i \tag{26}$$

with $\mathbf{s}_i = \mathbf{Y}_i\mathbf{Y}_{F_f}^T(\mathbf{Y}_{F_f}\mathbf{Y}_{F_f}^T)^{-1}\mathbf{G}_{F_f} - \mathbf{G}_i^T$ and $\beta_i = \mathbf{Y}_i(\mathbf{I} - \mathbf{Y}_{F_f}^T(\mathbf{Y}_{F_f}\mathbf{Y}_{F_f}^T)^{-1}\mathbf{Y}_{F_f})\mathbf{Y}_i^T$.

Note that the relationship in (24) is an inequality, which can be conservative. As a BAB method spends most of its time in evaluating nodes that cannot lead to the optimal solution, we use the computationally cheaper albeit weaker pruning criteria in this paper.

5. NUMERICAL EXAMPLES

To examine the efficiency of the proposed BAB algorithms developed in this work and listed in Table 1, numerical tests are conducted using randomly generated matrices and binary distillation column case study. All tests are conducted on a Windows XP SP2 notebook with an Intel® Core™ Duo Processor T2500 (2.0 GHz, 2MB L2 Cache, 667 MHz FSB) using MATLAB® R2008a.

Table 1. BAB programs for comparison

program	description
UP	upwards pruning (22)
DOWN	downwards pruning (23)
B^3	bidirectional BAB by combining (22) and (23)
PB^3	partially B^3 by combining (23) and (24)

5.1 Random tests

To evaluate the efficiency of the different BAB algorithms developed in this work, we consider selection of n_u out of $n_y = 36$ variables, where n_u varies between 1 and 35 with $n_d = 5$. Six random matrices are generated: three full matrices, $\mathbf{G}^y \in \mathbb{R}^{n_y \times n_u}$, $\mathbf{G}_d^y \in \mathbb{R}^{n_y \times n_d}$ and $\mathbf{J}_{ud} \in \mathbb{R}^{n_u \times n_d}$, and three diagonal matrices, $\mathbf{W}_e \in \mathbb{R}^{n_y \times n_y}$, $\mathbf{W}_d \in \mathbb{R}^{n_d \times n_d}$ and $\mathbf{J}_{uu} \in \mathbb{R}^{n_u \times n_u}$. The average computation time and number of nodes evaluated over the 100 random cases are summarized in Figure 2.

From Figure 2, it can be seen that all the developed algorithms (UP, DOWN and B^3) show much superior performance than the currently used brute force method. As one may expect, upwards pruning based algorithm (UP) shows better efficiency for problems involving selection of a few variables from a large candidate set, whilst downwards pruning based algorithm (DOWN) is more efficient for problems, where a few among many candidate variables need to be discarded to find the optimal solution. The solution times for the B^3 algorithm is similar to the better of UP and DOWN algorithms, however, its efficiency is insensitive to the kind of selection problem. Within 1000 seconds, both UP and DOWN algorithms can only handle problems with $n_u < 9$ or $n_y - n_u < 9$. For all cases,

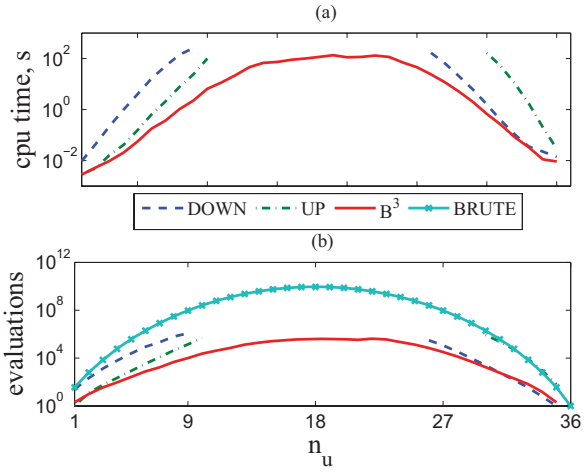


Fig. 2. Random test: (a) computation time and (b) number of nodes evaluated against n_u.

however, the B^3 algorithm exhibits superior efficiency and is able to solve the problem with $n_u = 18$ within 200 seconds.

5.2 Distillation column case study

To demonstrate the efficiency of the developed PB^3 algorithm, we consider self-optimizing control of a binary distillation column (Skogestad, 1997). The objective is to minimize the deviation of the distillate and bottoms composition from their nominal steady-state values in presence of disturbances in feed flow rate, feed composition and vapor fraction of feed. Two degrees of freedom (reflux and vapor boilup rates) are available and thus two CVs are required for implementation of self-optimizing control strategy. It is considered that the temperatures on 41 trays are measured with an accuracy of $\pm 0.5^o$ C. The combinatorial optimization problem involves selection of n out of 41 candidate measurements, whose combinations can be used as CVs. The reader is referred to Hori and Skogestad (2008) for further details of this case study.

The PB^3 algorithm is used to select the 10 best measurement combinations for $2 \leq n \leq 41$. The trade-off between the losses of the 10 best selections and n is shown in Figure 3(a). It can be seen that when $n \geq 14$, the loss is less than 0.075, which is close to the minimum loss (0.052) by using a combination of all 41 measurements. Furthermore, the reduction in loss is negligible, when combinations of more than 20 measurements are used.

Figures 3(b) and (c) show the computation time and number of node evaluations for PB^3 and DOWN algorithms. Overall, both algorithms are very efficient and are able to reduce the number of node evaluations by 5 to 6 orders of magnitude, as compared to the brute force search method. For example, to select 20 measurements from 41 candidates, evaluation of a single alternative requires about 0.15 ms on the specified notebook computer. Thus, a brute force search methods would take more than one year to evaluate all possible alternatives. However, both PB^3 and DOWN algorithms are able to solve this problem within 100 seconds. Hence, without algorithms developed here, it

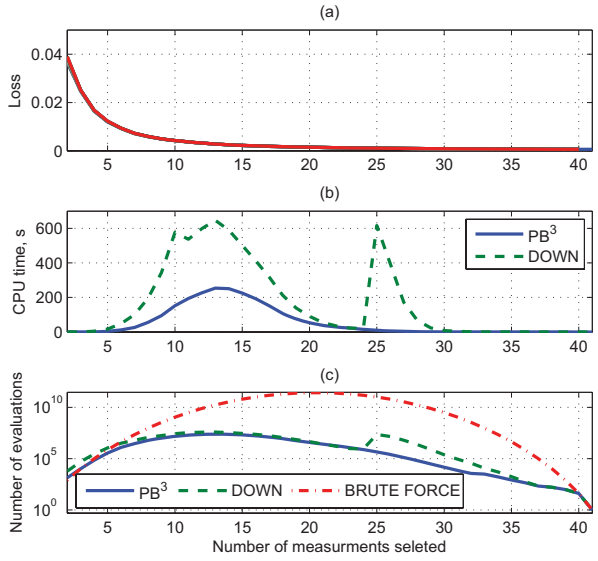


Fig. 3. (a) Average losses of 10-best measurement combinations against the number of measurements, (b) Comparison of computation time, and (c) Comparison of number of node evaluations

would be practically impossible to generate of the trade-off curve shown in Figure 3(a).

Due to the conservativeness of the pruning condition (24), the PB^3 algorithm is only able to reduce the number of node evaluations and hence computation time up to a factor of 2 for selection problems involving selection of a few measurements from a large candidate set. It is expected that a less conservative or fully upwards pruning rule would improve the efficiency, but the derivation of such a rule is currently an open problem.

6. CONCLUSIONS

In this paper, the concept of bidirectional branch and bound (BAB) proposed in Cao and Kariwala (2008) has been further developed for selection of controlled variables (CVs) using the local average loss minimization criterion for self-optimizing control (Kariwala et al., 2008). The numerical tests using randomly generated matrices and binary distillation column case study show that the number of evaluations for proposed algorithms is 4 to 5 orders of magnitude lower than the current practice of CV selection using brute force search.

The computational efficiency of the algorithms developed in this paper based on bidirectional pruning and branching principles and fast pruning algorithms is compatible to the BAB approach for CV selection based on minimum singular value (MSV) rule (Cao and Kariwala, 2008) and the local worst-case criterion (Kariwala and Cao, 2009). Despite the availability of the exact local criteria (the worst case and average loss), one of the apparent reasons for continued use of the approximate MSV rule is its computational efficiency. This work makes CV selection using the local average loss criterion computationally tractable so that it can be adopted as a standard tool for CV selection in the self-optimizing control framework.

While the algorithm for selection of individual measurements as CVs is fully bidirectional, the algorithm for selection of subset of measurements, whose combinations can be used as CVs, is only partially bidirectional. It is expected that the development of a fully bidirectional BAB algorithm for the latter problem would improve the computational efficiency further. Furthermore, the combination matrix H that minimizes average loss also minimizes worst-case loss (Kariwala et al., 2008). This super-optimality, however, only holds for a given subset of measurements and in general, different measurement subsets can be optimal for these two criteria. An extension of the bidirectional BAB algorithm to select CVs based on the bi-objective minimization of local worst-case and average losses for self-optimizing control is currently under consideration.

ACKNOWLEDGEMENTS

The first author gratefully acknowledges the financial support from Office of Finance, Nanyang Technological University, Singapore through grant no. RG42/06.

REFERENCES

Alstad, V., Skogestad, S., and Hori, E.S. (2009). Optimal measurement combinations as controlled variables. *J. Proc. Control*, 19(1), 138–148.

Cao, Y. and Kariwala, V. (2008). Bidirectional branch and bound for controlled variable selection: Part I. Principles and minimum singular value criterion. *Comput. Chem. Engng.*, 32(10), 2306–2319.

Halvorsen, I.J., Skogestad, S., Morud, J.C., and Alstad, V. (2003). Optimal selection of controlled variables. *Ind. Eng. Chem. Res.*, 42(14), 3273–3284.

Hori, E.S. and Skogestad, S. (2008). Selection of controlled variables: Maximum gain rule and combination of measurements. *Ind. Eng. Chem. Res.*, 47(23), 9465–9471.

Horn, R.A. and Johnson, C.R. (1985). *Matrix Analysis*. Cambridge University Press, Cambridge, UK.

Kariwala, V. (2007). Optimal measurement combination for local self-optimizing control. *Ind. Eng. Chem. Res.*, 46(11), 3629–3634.

Kariwala, V. and Cao, Y. (2009). Bidirectional branch and bound for controlled variable selection: Part II. Exact local method for self-optimizing control. *Comput. Chem. Eng.*, Accepted for publication.

Kariwala, V., Cao, Y., and Janardhanan, S. (2008). Local self-optimizing control with average loss minimization. *Ind. Eng. Chem. Res.*, 47(4), 1150–1158.

Narendra, P. and Fukunaga, K. (1977). A branch and bound algorithm for feature subset selection. *IEEE Trans. Comput.*, C-26, 917–922.

Skogestad, S. (1997). Dynamics and control of distillation columns - A tutorial introduction. *Trans. IChemE Part A*, 75, 539–562.

Skogestad, S. (2000). Plantwide control: The search for the self-optimizing control structure. *J. Proc. Control*, 10(5), 487–507.

Skogestad, S. and Postlethwaite, I. (1996). *Multivariable Feedback Control: Analysis and Design*. John Wiley & sons, Chichester, UK, 1st edition.

Plantwide control of fruit concentrate production

Mark van Dijk, Sander Dubbelman, Peter Bongers *

**Unilever Research Vlaardingen Netherlands, Process Systems Engineering.*
(E-mail: mark-van.dijk@unilever.com).

Abstract: Fruit concentrates are key ingredients in many fruit based Unilever products. We have designed a novel continuous fruit concentration process involving a decanter, an evaporator and a recombination process. In order to ensure best product quality and highest capacity a methodology for control structure design for complete processing plants (plantwide control) was applied. This included defining the control objectives, degrees of freedom analysis, definition of inventory and production rate control and the development of a non-linear dynamic model. Using the methodology, control alternatives were systematically analyzed and eliminated. The chosen control structure was successfully applied and implemented in a Unilever factory.

Keywords: Control structure design; Plantwide control; Dynamic control model; Decanter; Evaporator.

1. INTRODUCTION

Plantwide control is viewed as a strong methodology to design effective control structures in complex food plants.[1]

This paper presents the conceptual process control design of a novel fruit concentration method using a separation and recombination step. It involves the development of a dynamic control model of the system, the design of a suitable control strategy with the help of this model, the implementation of the control strategy in an actual factory process control system and evaluation of its performance.

1.1 Background

Traditionally fruit purees are produced by concentrating the fruit pulp in a forced recirculation evaporator, which are known to be detrimental to product quality because of their large residence time and high temperatures.

We have developed a novel fruit concentration process involving the use of a separation step to separate the fruit pulp into a cake fraction and a serum fraction. Because of the absence of fibres in the serum fraction, the serum fraction can be concentrated at lower temperatures, thus maintaining its freshness. Simultaneously, the fibres are not exposed to high shear and will therefore better maintain their water-binding properties.

A draw-back of the novel fruit concentration process is that the cake fraction and the concentrated serum fraction need to be recombined into a homogeneous fruit paste in a controlled way. At the same time, the Brix and viscosity of the resulting fruit concentrate need to be maintained within normal food standards. For this reason, a systematic approach is required in order to design an effective control strategy.

1.2 Approach

Plantwide control is a concept or methodology to systematically build control structures of large, continuous processes with complex interactions [1, 2, 3]. Process modelling and simulation form an important aspect of the method. It follows a series of logical steps which are easy to apply in practice:

1. Definition of Operational Control Objectives
2. Manipulated Variables and Degrees Of Freedom
3. Primary Controlled Variables
4. Production Rate
5. Regulatory Control Layer
6. Supervisory Control Layer
7. Optimization Layer
8. Validation

We applied most steps of this methodology in conjunction with a conceptual process design methodology [4] to ensure that the integration of process, equipment and control design is optimal. This paper will focus on the first five steps. In chapter 2 we will go systematically through steps 1 to 5 given above based on the process given in figure 2.1. Next, we will present the assumptions behind the underlying mathematical model that we used to determine the best control strategy. Finally we will present the results from the industrial process that we implemented in one of our factories.

2. PLANTWIDE CONTROL PROCEDURE

2.1 STEP 1: Overall control objective. Identify operational constraints.

The process control aims at producing high grade, constant quality fruit paste. Good quality paste has a nice red

[1] This is in contrast with plantwide automation which addresses the design and implementation of Industrial Process Control and Automation Systems (IPCAS).

colour, constant Brix (sugars) and Bostwick (viscosity) values and also good taste and flavour. In order to achieve this, it is necessary to control the product flow throughout the plant smoothly and to minimise the hold-up time of product in the system. The following process and control system requirements were considered:

Process Requirements

Maximization of assets. Maximization of plant output is a key objective since fruit concentration processes are mostly seasonal continuous operations. It is therefore important that the bottleneck is 100% utilised. In our case, the evaporator is the bottleneck of the process.

Process reliability. There are many factors in the process not directly related to the control system or strategy, which influence the control performance. A major factor is the overall factory reliability (equipment and utilities). The novel process is more complex and especially vulnerable for unplanned stoppages. Both factory reliability and the control system robustness towards unplanned stoppages are essential aspects to consider in the design process.

Equipment restrictions. It is industry practice to operate decanters with a constant in-flow in order to get a stable liquid/solid separation. It is recommended that feed flow variations are less than ±20%. It is also industry practise to operate forced recirculation evaporators with a constant steam supply (constant evaporation rate).

Fruit quality fluctuations. The fibre- and sugar content of raw fruit juice is known to vary up to ±20% on an hourly basis. The fibre- and sugar content vary independently.

Control system requirements

End-product variability Brix. The Brix boundaries should be less than 1.0 Brix from controller setpoint for commercial purposes.

End-product variability Bostwick. The Bostwick should be the same or lower as normal variations in standard paste (typically +/- 0.5 cm). This implies that there should not be *more* variation in the ratio of sugars to fibers in the puree as in the original fruit.

Simple control. The control strategy must be able to be implemented in the SIEMENS Control System and must be relatively easy to understand for the operators. For this reason, the control strategy is based on standard PID-controllers.

2.2 STEP 2: Manipulated variables and degrees of freedom.

According to the principles by Skogestad, we will identify dynamic and steady-state Degrees Of Freedom (DOF) based on the process given in figure 2.1 according to the equation below:

$$N_{ss} = N_m - (N_{0m} + N_{0y})$$

In which:

N_{ss}	Number of steady state DOFs
N_m	Number of dynamic (control) DOFs (valves, pumps)
N_{0m}	Number of manipulated input variables with no steady state effect
N_{0y}	Number of output variables with no steady state effect (product levels)

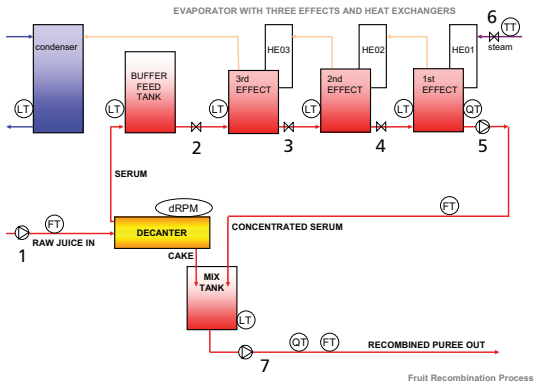


Figure 2.1: Process Flow Diagram of novel fruit concentration process with Degrees Of Freedom (DOFs).

As can be seen from figure 2.1, 7 dynamic DOFs (N_m) can be identified. Since the process consists of 5 tanks, there are 5 tank levels that need to be controlled (N_{0y}), but that do not contribute to the steady state mass- and composition balance. This means that there are two remaining DOFs (N_{ss}) to control the whole process. This is identical to an existing fruit concentrate process in which the two DOFs are used to control the Brix of the puree and the production rate. However we know that the dynamics of the novel process have changed, because we have combined a slow evaporation process with a fast separation process in one additional process step (the mix tank). Therefore a dynamic model is essential to evaluate the new process dynamics and possible process control structures.

2.3. STEP 3: Primary controlled variables.

Skogestad states that the primary variable to control is the active constraint. According to our control objectives, this is overall production rate as set by the evaporator. Another key primary controlled variable is the Brix of the resulting fruit puree. The Brix is given by the following equation:

$$\phi_{puree} \cdot Brix_{puree} = \phi_{cake} \cdot Brix_{cake} + \phi_{concentrated\ serum} \cdot Brix_{concentrated\ serum}$$

in which:

ϕ_{puree}	Flow of puree [kg/s]
$Brix_{puree}$	Brix of puree [Brix]
ϕ_{cake}	Flow of cake [kg/s]
$Brix_{cake}$	Brix of cake [Brix]
$\phi_{concentrated\ serum}$	Flow of concentrate serum [kg/s]
$Brix_{concentrated\ serum}$	Brix of concentrate serum [Brix]

The flow rate of cake is mainly driven by the amount of fibres in the fruit juice as well as the way the decanter is operated (level of drying of the cake), whereas the Brix of the cake is driven by the fruit variety, ripeness and other agronomical factors. Both Brix and flow rate of concentrated serum are highly dependent on the way that

the evaporator is operated and controlled. The chance of product that is out of specification is high since it is dependent on four variables.

2.4 STEP 4: Set of production rate.

The choice where to set the production rate determines the structure of the remaining inventory (level) control system. The production rate should be set at the (dynamic) bottleneck of the process as explained in section 2.3. For the traditional fruit concentrate process, the production rate is determined by the slow evaporator (dynamic bottle-neck). The evaporator is operated at a fixed steam pressure (constant evaporation rate). The up-stream and down-stream processes follow the evaporation rate. Since the end product Brix is controlled through manipulating the evaporator out-feed pump, this means the product flow through all unit operations is changing constantly dependent on the incoming Brix of the raw juice.

On the other hand, it is industry experience that decanters require a constant feed (within 20% of the main flow) to have the best performance. This requires controlling the flow towards the decanter and setting the production rate here, through a flow controller.

In order to control the level in the evaporator feed tank, therefore two options do exist:

A. **Control the level via manipulation of the steam supply** (thus via manipulation of the evaporation rate). No constant steam supply means that the evaporator cannot be operated at constant evaporation rate and temperatures. The potential negative impact should be evaluated and preferably minimized.

B. **Control the level via manipulation of the inflow to the decanter.** It is clear that in this scenario no constant inflow to the decanter can be guaranteed and that the potential negative impact on separation performance should be evaluated and minimized. It is noted that controlling the level through decanter serum outflow will disturb the separation process and therefore is not feasible.

Either scenario clearly forced us to deviate from the industry practice w.r.t equipment operation.

Already at this stage of the methodology we decided to develop a dynamic model based on the following arguments:

- The choice of setting the production rate highly influences the dynamics of the overall process
- There is a direct correlation between productivity and end-product composition in concentration processes
- The recombination process is highly vulnerable to variations
- We deviate from the industry standard w.r.t. equipment operation for either the decanter or the evaporator.
- The design of the control system will influence the process design, like the design of the recombination tank.

The model must involve the complete regulatory control layer in order to decide on the best control strategy. In section 2.5, we will provide the results from the model simulations and we will demonstrate how we used this to evaluate the various control strategies.

2.5. STEP 5, 6 and 8: Regulatory and Supervisory Control layers and Validation.

Section 2.4 showed that the production rate could not be defined because of the strong interactions between production rate, inventory control and product composition. It was also shown that there are two DOFs. With these two DOFs, a steady state analysis demonstrates that we can control the production rate and the Brix of the fruit puree in the following way:

1. Puree Brix control through manipulation of the concentrated serum Brix. Steady state mass- and composition balances show that an increase in concentrated serum Brix will increase the Brix of the resulting puree. Based on this, a cascade control loop can be designed. In this way, no extra DOF needs to be created and no additional equipment is required.

However, if the dynamics of this control structure prove to be unsuitable to control the Brix of the fruit puree, one additional DOF will be required in order to make the process controllable. This DOF should have sufficiently fast response time and a sufficient process gain. We have designed two alternative ways to create one extra DOF:

2. Puree Brix control through manipulation of decanter settings. By changing the differential speed of the screw inside the decanter [5] the amount of juice incorporated inside the cake can be controlled. Reducing the differential speed will send less water via the cake to the recombination tank and the Brix of the resulting puree will increase.

3. Puree Brix control through addition of juice to recombination tank. A third stream of juice can be added to the recombination tank to dilute a slightly over-concentrated puree to the specified composition.

All scenarios are summarized in table 2.1.

Table 2.1: Overview of control scenarios.

Production rate / Brix control	Extra DOF created?	A. Control the level via steam supply	B. Control the level via inflow to the decanter.
1. Puree Brix control through manipulation of the concentrated serum Brix	No	A1	B1
2. Puree Brix control through manipulation of decanter settings.	Yes	A2	B2
3. Puree Brix control through addition of juice to recombination tank	Yes	A3	B3

Despite the fact that scenarios A1 and B1 are feasible based on steady state, an inverse response occurs initially: Upon increasing of the concentrated serum Brix (see figure 2.2 at t=10 hours), the outflow pump first slows down in order to increase the residence time in the first effect. The result is an inverse response of the puree Brix. This clearly demonstrates that in concentration processes, flow and concentration level are inversely proportional.

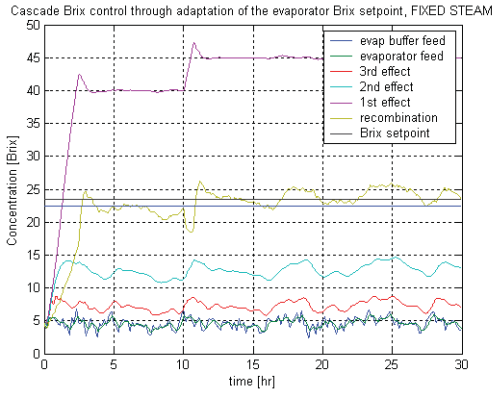


Figure 2.2: Dynamic model simulation of scenario A1. The inverse response in puree Brix (denoted in yellow-green as 'recombination') at t=10 hours is clearly visible.

The conclusion is that the additional DOF is required (Scenarios A2, B2, A3 and B3). In order to decide on the optimal control strategy, we will first evaluate the A2 and B2-strategies against the A3 and B3-strategies.

For strategies A2 and B2 the dynamic behaviour and operating window of the decanter needs to be known. Based on step response measurements on the serum exiting the decanter (see figure 2.3), we estimated that the response time of the cake fraction upon changes in differential speed is in the order of 300 seconds (three times the response time of the serum fraction).

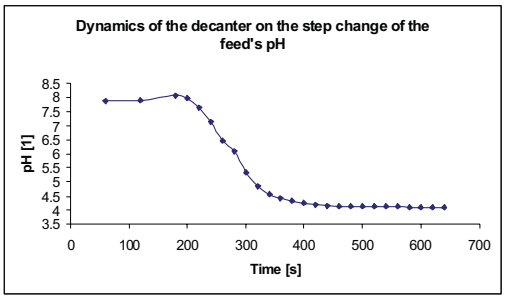


Figure 2.3: Response time of decanter serum exit on a step change in pH via citric acid addition (at 170 s).

Figure 2.4 shows the response behaviour of the solids concentration in the cake exiting the decanter as a function of changes in the differential speed (courtesy of GEA Westfalia). It can be seen that there is a strong response, but only in a small operating window. Also can be seen that the machine operation is highly non-linear.

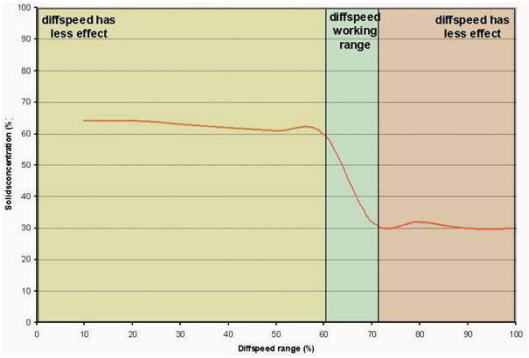


Figure 2.4: Response of solids concentration in the cake from a decanter. Courtesy of GEA Westfalia GmbH.

The results of the simulations with the above time delay and response behaviour are shown for scenario A2 in figure 2.5. As can be seen the fruit puree Brix cannot be kept within the desired range. Scenario B2 (not shown) behaved in a similar way. This means that the decanter is too slow to be used in a control loop, thus rendering strategies A2 and B2 ineffective.

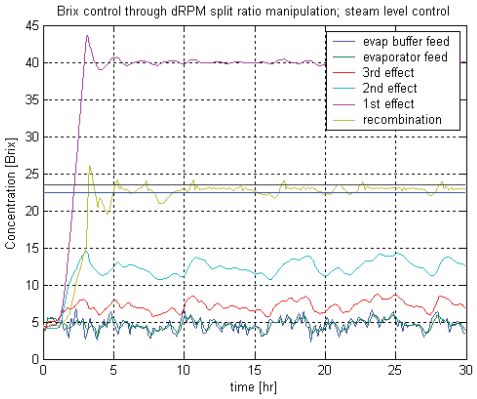


Figure 2.5: Dynamic model simulation of scenario A2. Fruit puree Brix control through manipulation of decanter settings. It can be seen that the Brix of the fruit puree (denoted in yellow-green as 'recombination') cannot be maintained within the desired range of ±1 Brix around setpoint.

We now need to decide between the scenarios A3 and B3. For a stable process, it is important that both concentrated serum and cake are recombined according to its natural ratio. This means that there cannot be an excess flow of either cake or concentrated serum in the mix tank. In this way no changes in hold-up volume of either the cake fraction or the serum fraction should occur. The only location where significant changes in hold-up volume can occur is the evaporator feed tank.

Figure 2.6 shows the level in the evaporator feed tank for scenarios A3 and B3. It can be seen that both control

strategies can control the level in this tank, but control strategy A3 requires a much longer time upon start-up to reach a stable level (> 5 hours). This means that this strategy is very slow due to the slow response of the evaporator upon changes in steam supply. For this reason, this control strategy is very vulnerable to disturbances and not recommended.

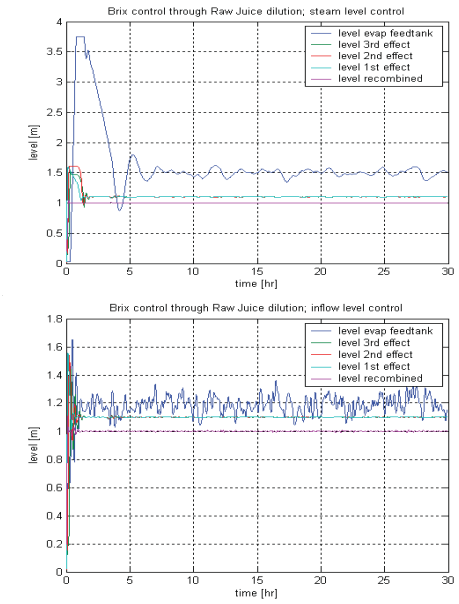
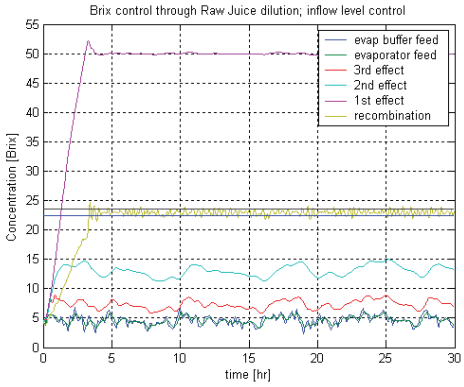


Figure 2.6: Dynamic model simulation of scenarios A3 (top) and B3 (bottom). It can be seen that the time delay to reach a setpoint of the evaporator feed tank is in the case of scenario A3 is in the order of 5 hours and in scenario B3 in the order of 1 hour.

Scenario B3 gave the right dynamic behaviour to control the evaporator feed tank, allowing for a proper recombination of concentrated serum and cake. Figure 2.7 demonstrates that this strategy is also able to control the Brix of the fruit puree. Also, it can be seen that the variation of the ratio of sugars to fibers is within variations found in the raw material. Thus control strategy B3 is most suitable for this process.



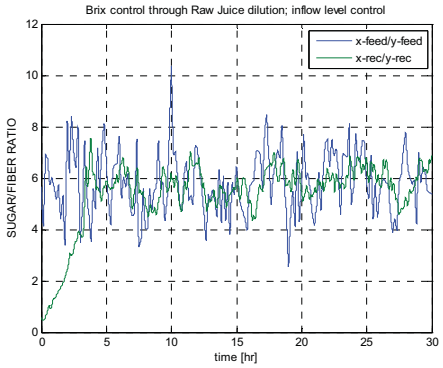


Figure 2.7: Dynamic model simulation of scenario of control strategy B3. It can be seen that the Brix of the fruit puree (denoted in yellow-green as 'recombination') can be maintained within the desired range of ±1 Brix around setpoint (left bottom). The ratio of sugars to fibers in the feed (x-feed/y-feed) and in the fruit puree (x-rec/y-rec) are shown above.

Proper tuning of the level control loop of the evaporator feed tank further contributes to the overall process stability. In order to minimize fast fluctuations in decanter inflow, it was decided to use a P-algorithm and to set the gain value relatively low. In this way, the evaporator feed tank can be used as a real buffer without impacting on the ratio of sugars to fibers. Also feed fluctuations to the decanter are minimised.

The resulting process flow diagram for strategy B3 with all sensors, actuators and control loops is given in figure 2.8.

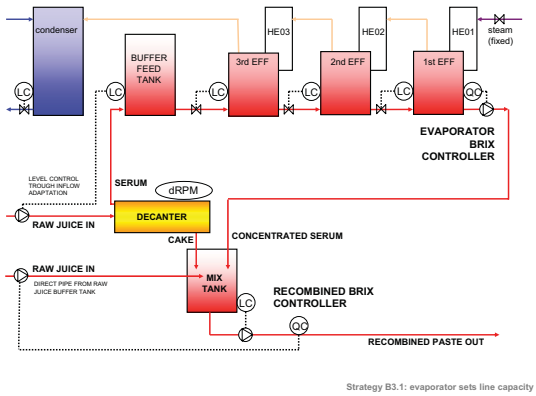


Figure 2.8: Process flow diagram with for strategy B3 with sensors, actuators and the final control structure.

3. DYNAMIC MODEL

We built a non-linear dynamic model in Matlab/Simulink [6] based on:

- Dynamic mass- and composition balance for all tanks
- Dynamic heat balance for all heat exchangers with the heat transfer coefficient estimated based on real-time factory data
- Steady state equations for the decanter with a first order transfer function with lag time
- Band-limited white noise in feed sugar and fibre levels of ±20% around the average
- PID-feedback control

4. IMPLEMENTATION INTO THE FACTORY

4.1 IPCAS implementation

The process and control strategy as defined before was implemented in a Unilever factory. To this purpose a "User Requirements Specification of the Industrial Process Control and Automation System (IPCAS)" was written. It specifies the Industrial Process Control System (PLC/SCADA process computer hardware and software), instrumentation and installation infrastructure.

4.2 Evaluation of control strategy B3 under factory conditions

The performance of the selected control strategy that was implemented in the SIEMENS PLC/SCADA system was evaluated during actual production.

Figure 4.1 shows that for the given day, the whole production was within specifications. During one whole season, the average Brix was 22.92 (with a setpoint of 23) with a standard deviation of only 0.3.

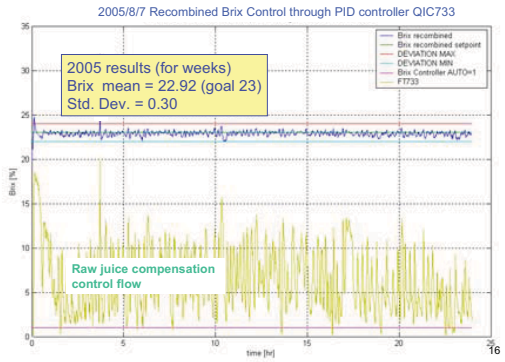


Figure 4.1: Factory data of fruit puree Brix (named QIC 733 in factory control system). Also the flow rate (named FT733 in the factory control system; raw juice compensation control flow) of the added juice is depicted. This flow rate should be multiplied by 100 and is expressed in kg/hour.

5. CONCLUSIONS

In this paper we presented a novel fruit concentrate production process involving a decanter, an evaporator and a recombination process. The choice for this process results in complex, non-linear, process dynamics. Such processes can be difficult to control and a systematic methodology was required.

We demonstrated that a strategy in which the Brix is controlled via addition of a third juice stream was the best choice for the given process. Evaluation of the control strategy under real factory conditions showed that the control strategy is very robust and that end-product specifications are met.

This case demonstrated to us that it is important to integrate control strategy design in an early stage with process and equipment design. Furthermore we required a non-linear dynamic model to understand the complex dynamics of the process and to design an appropriate control strategy.

REFERENCES

[1] Skogestad S., ″Control Structure Design for Complete Chemical Plants″, Computers and Chemical Engineering, 28 (2004), pp. 219-234.

[2] Luyben M.L., Tyreus B.D., Luyben W.L., ″Plantwide Control Design Procedure″, AIChE Journal, 43 (1997), pp. 3161-3174.

[3] E. M. Vasbinder, K. A. Hoo and U. Mann, Computer Aided Chemical Engineering, Volume 17, 2004, Pages 375-400 ; ″Synthesis of plantwide control structures using a decision-based methodology″.

[4] Douglas J.M., ″Conceptual Design of Chemical Processes, McGraw-Hill″, New York (1988).

[5] Records A., Sutherland K. ″Decanter Centrifuge Handbook″, Elsevier, Oxford (2001).

[6] Matlab/Simulink, Dynamic modelling and simulation software. The MathWorks, Inc.

Emerging Methods and Technologies

Oral Session

Monitoring, Analysis and Diagnosis of Distributed Processes with Agent-Based Systems ⋆

Ali Çinar * **Sinem Perk** * **Fouad Teymour** * **Michael North** **
Eric Tatara ** **Mark Altaweel** **

* *Department of Chemical and Biological Engineering, Illinois Institute of Technology, Chicago, IL 60616 USA (e-mail: perksin@iit.edu)*
** *Argonne National Laboratory, Argonne, IL 60439 USA (e-mail: north@anl.gov)*

Abstract: Multiagent systems provide a powerful framework for developing real-time process supervision and control systems for distributed and networked processes by automating adaptability and situation-dependent rearrangement of confidence to specific monitoring and diagnosis techniques. An agent-based framework for monitoring, analysis, diagnosis, and control with agent-based systems (MADCABS) is developed and tested by using detailed models of chemical reactor networks. MADCABS is composed of three main hierarchical layers, the physical communication layer, the supervision layer and the agent management layer. The supervision layer consists of agents and methods for data preprocessing, process monitoring, fault diagnosis, and control. The agent management layer conducts the assessment of agent performances to assign the priorities for selecting the most useful methods of process supervision for specific types of situations. The paper illustrates the operation of MADCABS for monitoring and fault detection.

Keywords: Multi-agent systems, process monitoring, fault detection, fault diagnosis, process supervision, process control, distributed systems, distributed artificial intelligence, autocatalytic reactions.

1. INTRODUCTION

Multi-layered and adaptive multiagent systems (MAS) provide a powerful framework for developing a new generation of real-time supervision and control systems for distributed and networked processes. The strategy, techniques and tools are being developed at IIT for monitoring, analysis, diagnosis, and control with agent-based systems (MADCABS) that automates knowledge extraction from data, analysis, and decision making. This distributed artificial intelligence framework is expected to enable the consideration of novel configurations for manufacturing such as distributed reactor networks to produce high-value-added specialty chemicals.

There are strong reasons for distributing the activities and intelligence in software for supervision of distributed process operations:

- The complex layout of a manufacturing process yields a problem that is physically distributed,
- The supervision problem is distributed and heterogeneous in functional terms,
- The complexity of the supervision problem dictates a local point of view that contributes to the development of system-wide decisions that may force reexamination of local decisions,
- The supervision system must be able to adapt to changes in the structure or environment of the supervised process or network.

Agents are capable of acting, communicating with other agents, perceiving their environment, and determining behavior to satisfy their objectives. They are endowed with autonomy and they possess resources. However, the MAS framework offers challenges as well: Agents have only partial information about their environments, they may act "selfishly" or initiate actions that may conflict with actions of other agents. This may lead to undesirable or harmful behavior in MADCABS or the supervised process, compromising its profitability and safety.

The nature of the supervision problem dictates the use of multiple layers of agents where lower-level agents perform local well-defined tasks such as information validation from sensors and higher-level agents perform more global tasks over wider regions of the supervised system. Several agents can be used to perform a specific task, each using different methods to enable not only decision by consensus-building but also to reduce the influence of the weaker methods over time. Intelligence and adaptation is provided both at agent and at system level.

MADCABS is composed of three main hierarchical layers, the *physical communication* layer, the *process supervision* layer and the *agent management* layer. The physical layer is where two-way information communication between the

⋆ This work is supported by the National Science Foundation CTS-0325378 of the ITR program.

process and MADCABS takes place. Process information, such as the flowchart of the process is mapped to MADCABS through the physical communication layer. The supervision layer consists of agents and methods for data preprocessing, process monitoring, fault diagnosis, and control. Data preprocessing agents filter the process data, check for outliers and missing data, and provide estimates for them. Monitoring agents detect deviations from normal operation and trigger the fault detection and diagnosis (FDD) agents. When abnormal process operation is validated, FDD is carried out using contribution plots, statistical methods, and process knowledge. Control agents range from simple local PI controllers to decentralized plant-wide grade transition agents. Some agents collaborate to help each other, while others work in a competing manner to satisfy a global objective. The performances of different agents and methods are evaluated in the topmost agent management layer. The agent management layer is responsible for selecting the best performing agents for process monitoring, fault detection and diagnosis, and process control for the current operating conditions. The assessment of agent performances guide the priorities assigned to select the most useful methods of process supervision for specific types of situations.

In this paper, MADCABS modules for monitoring and fault detection, and the information flow among them is discussed. The paper focuses on the architecture and functionality of MADCABS, the automated tools for assessing the success of its various functions, the redundancies in MADCABS, and the adaptation of MADCABS based on the current state of process operations. The communication and cooperation between different MADCABS modules is demonstrated with case studies using autocatalytic CSTR networks. The capabilities of MADCABS in detecting and diagnosing various types of faults are shown.

2. PROCESS MONITORING, FAULT DETECTION AND DIAGNOSIS

2.1 Statistical Process Monitoring Techniques

Multivariate statistical process monitoring (SPM) techniques are used in this study. Multivariate techniques that extract the correlation among variables based on principal component analysis (PCA) provide the basic tool for monitoring continuous processes (Kourti and MacGregor (1996); Cinar et al. (2007); Jackson (1980)). PCA is a multivariate projection method that extracts strong correlations among the variables in a data set, and based on that information defines a new orthogonal coordinate space where the coordinate axes are the highest variance directions. For chemical processes, where large highly-correlated process datasets need to be monitored, singularity problems may arise. PCA is well-suited for reducing the dimensionality of the data and capturing the essential information in the data.

For large processes that involve many processing units and many process variables with different correlation structures, a single PCA model for the whole process may not give sufficient explanation about the process behavior, may provide unreliable information based on many false and missed alarms, and may have difficulty localizing the source cause among so many variables when a fault is detected. Multiblock methods have been proposed in literature for large processes, where the process can be separated into meaningful process blocks, to increase the efficiency and interpretability of the statistical monitoring model. Algorithms to handle multiple data blocks include hierarchical PCA (HPCA) and consensus PCA(CPCA) (Qin et al. (2001); Wangen and Kowalski (1988); Westerhuis et al. (1998); Wold et al. (1996)). Multiblock algorithms enable monitoring of the process both locally and globally. The CPCA method is designed for comparing several blocks of descriptor variables measured on the same objects (Wold et al. (1996); Westerhuis et al. (1998)).

Dynamic PCA (DPCA) is an extension of conventional PCA to deal with multivariate process data that is correlated in time, using a time-lag shift method (Ku et al. (1995)). The flow of action, namely, monitoring, fault detection and diagnosis, is the same for all SPM methodologies independent of which monitoring algorithm is employed.

2.2 Fault Detection, Monitoring and Diagnosis Framework in MADCABS

MADCABS is written in Java, using Repast Simphony as the agent building platform (ROAD (2005)). Among the important features of Repast that MADCABS uses are its object oriented structure, scheduling tools and its built-in automated Monte Carlo simulation framework. Repast also allows users to change, add, delete agents in run time.

In Repast Simphony, a *context* is defined as a container where the agents reside. There are three contexts for monitoring, fault detection and diagnosis agents in MADCABS. The communications between different agents in these contexts are shown in Figure 1. Statistical models are built by the monitoring agents. The fault detection agents, which are the monitoring statistics, assign themselves to the fault detection organizer agents responsible for each subsystem that is monitored. A fault is flagged when a consensus is formed among different fault detection agents on the existence of a fault in the system. The fault flag triggers the diagnosis agent. Diagnosis agent uses information from the neighboring fault detection organizers, finds the most contributing process variables to the fault on the faulty subsystem, and investigates the potential reasons behind the fault.

Monitoring Agents. The monitoring agents are PCAStarter, DPCAStarter and MultiblockStarter agents and a monitoring organizer (Figure 2). After the system reaches steady state and a sufficient amount of normal operation data is available, monitoring starters are scheduled to form models. For all process units, a local statistical model is built using both PCA and DPCA. In the distributed framework, this is achieved by creating as many PCAStarters and DPCAStarters as the number of operating units. Each PCAStarter agent builds a separate PCA model since the data and the PC number that is used to build the models are different for each unit. DPCAStarters work identical to PCAStarters. There is only one MultiblockStarter for the whole process. In this case, the data blocks consist of data from each operating unit. The MultiblockStarter forms a single multiblock model, which

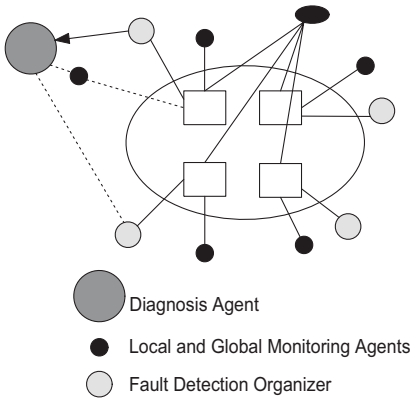


Fig. 1. Monitoring, fault detection and diagnosis agents in a four reactor network

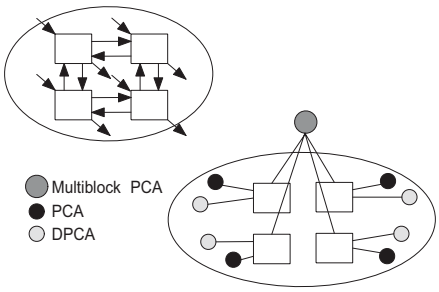


Fig. 2. Monitoring agents. The upper figure represents a four reactor network, which is simplified in the lower figure.

enables the monitoring of the process both locally and globally through block statistics and super statistics. Since there is only one model, the size retained in the model is the same for each block.

The starter agents are subclasses of MonitoringStarterParent class. Consequently, each of the starters extend some of the characteristics from the parent class and they also have class specific properties. The methods in each starter such as the buildModel and startProjection are common methods inherited from the parent, as well as the user-specified parameters. The number of principal components to be retained in the model is a superclass variable and it is overridden in each child class. Each model generates two monitoring statistics for each subsystem, a T^2 statistic and an SPE statistic. Three monitoring methods generate a total of six fault detection agents for each subsystem. The fault detection agents are contained in the fault detection context.

Fault Detection Agents. The fault detection agents, which are the T^2 and SPE statistics for each subsystem, assign themselves to the fault detection organizer of their subsystem (Figure 3). The fault detection organizer is responsible for keeping count of its fault detection agents, declaring consensus fault, keeping history of the performances of different fault detection agents under different fault scenarios, and in case of a consensus fault decision, triggering the diagnosis agent.

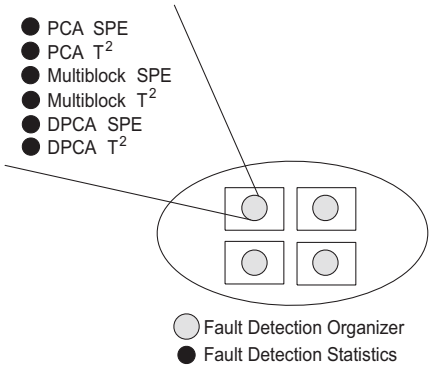


Fig. 3. Fault detection agents.

Statistics values and confidence limits are among the class variables for each fault detection agent. If the value of the statistic goes outside of the limits, the agent flags a fault. Fault detection organizer keeps track of all the fault flags given by its statistics. There are several criteria to form a consensus among different fault detection agents. The simplest would be to flag a fault, if the majority of the fault detection agents are flagging fault. This would require four of the six agents to flag a fault in order to declare that there is a fault in the unit. In the following, this strategy will be referred to as the "*number weighted*" consensus criteria.

Another criterion is based on the performances of fault detection agents over time, and based on their reliability, their decision is given more weight compared to less reliable fault detection agents. This is referred to as the "*reliability weighted*" consensus criteria. At each time point, when a new observation is available and new monitoring statistics are calculated, fault detection agents either detect a fault or not. Based on their decisions, they are given an instantaneous performance reward. The rewarding strategy is designed so that a missed alarm is penalized the most and the correct detection of fault is rewarded the most. A set of instantaneous performance rewards or penalties is given in Table 1, where the rows show the consensus and columns show the individual agent decisions. If the fault detection agent flags a fault but the consensus decision indicates otherwise, the agent is penalized for a false alarm. If the agent does not flag a fault, but the consensus flags a fault, then the agent is penalized for a missed alarm. A missed alarm is considered to be worse than a false alarm, and this is reflected in the instantaneous performance calculations. The instantaneous performances are summed in time for each detection agent, and makes up the accumulated performances. The reliability of an agent is determined by the accumulated performance values divided by the total accumulated performance value of all agents in that unit. The reliability weights are then considered in the consensus decision making.

Table 1. Instantaneous performance rewards

	Not faulty	Faulty
Not faulty	0.5	-0.5
Faulty	-1	1

The challenging problem of SPM methods is the missed and false alarm rates. For some cases, where the fault is

diffusing in the process and affecting the neighboring units and also with minor faults, the consensus flag may be oscillatory. This oscillation affects the performance mechanism in an undesired way such that an agent that has been flagging fault in the oscillatory period may not be reliable enough at that point to affect the consensus decision, and it will be penalized for flagging fault although the flag was right. Or an insensitive method could be rewarded if it did not flag the fault and again this would affect the consensus in an undesired way. In order to prevent these, the performances of agents are updated after fault episodes. A fault episode starts when a consensus fault is flagged. And the episode continues until no consensus fault is flagged for eight consecutive time points. At that point, looking back in history, the performances of agents are updated.

In order to design an automated fault detection framework, where the decisions of fault detection and diagnosis highly influence the succeeding tasks, reliability of the decisions is very important. Some of the monitoring methods may perform better than the others for various states of the process. Use of agent-based cooperation between different methods that are competing for the same task results in better overall performance than if these methods were used independently. Several monitoring agents have been implemented in MADCABS to provide diversity. The aim is to design an automated fault detection framework that can detect the faults on time, and that gives fewer false and missed alarms than if the monitoring methods were used independently. Comparison of different combinations of monitoring methods and the false and missed alarm rates of the corresponding monitoring statistics indicates that cooperation among agents improved the false and missed alarm rates.

Table 2. False and missed alarm summary of different fault detection agent combinations, using reliability weight condition

Agent	Missed Alarm	False Alarm
PCA	1.09	0.17
Multiblock PCA	20.98	0.01
DPCA	0.18	0.02
PCA-Multiblock PCA	1.21	0.06
DPCA-Multiblock PCA	0.75	0.07
PCA-DPCA	0	0.03
ALL	0	0.03

Diagnosis Agent. Diagnosis agent works in an event-driven way. It is activated when a consensus fault is flagged by a fault detection organizer. The responsibility of the diagnosis agent is to investigate the type and severity of the fault under consideration. Contribution plots are used as a diagnostic tool. The contributions of process variables to each fault detection statistic are calculated for different monitoring methods. The contribution plots do not indicate the source cause of the fault, but identify the variables that have contributed to the inflation in the SPM statistic. The diagnosis agent performs contribution plot analysis and determines the variables that inflated the SPM statistic that went out-of-control. At this point a sequence of events is activated. First, the most contributing variable to each statistic is chosen by checking if the variable contribution value is beyond the 3-*sigma* confidence limits and if it makes up a significant amount of the contributions. Each fault detection agent identifies their most contributing variables, and the common top most contributor to all is identified. That variable is then eliminated from the monitoring model data matrix and a new statistical model is built using the remaining variables. The aim is to detect if the fault is a sensor fault or a process fault. The assumption that is made here is, that a process fault usually has a fault signature and is realized in more than one variable. On the other hand, a sensor fault, especially a single sensor fault does not affect the other variables if it is not being used for control.

If the new observation, after the variable is eliminated, is in-control with the new model, it is declared as a potential sensor fault, however, the projection onto the new model continues in parallel to check if it will turn out to be a process fault later since some of the minor process faults can be misinterpreted as sensor faults in the beginning. If the new observation is not in-control with the new model, this means other variables have also been affected from the fault, and it is declared as a potential process fault. In addition to discriminating the types of faults, diagnosis agent estimates the severity of the fault by looking at how much the variable contributions have gone outside the confidence limits, how many of the neighboring fault detection organizer are signaling fault and how many of the fault detection statistics in the unit have identified the same fault signature.

3. MONITORING AND FAULT DETECTION OF A REACTOR NETWORK

3.1 Autocatalytic CSTR Network

Reactor networks hosting multiple species have a very complex behavior (Figure 2). As the number of steady states of the network increases, autocatalytic species are allowed to exist in the network that would otherwise not exist in a single CSTR. The cubic autocatalytic reaction for a single autocatalytic species is

$$R + 2P \rightarrow 3P \quad \text{and} \quad P \rightarrow D \tag{1}$$

R is the resource concentration, P is the species concentration, D is a dead species. The reaction rate for the first reaction, species growth rate constant, is k and k_d is the species death rate constant. The feed flow rates and interconnection flow rates are treated as manipulated variables. Each reactor has an inlet and outlet flow. The resource concentration in each reactor along with species concentrations is also available.

MADCABS is designed to work with process data from a real process plant or a process simulator. The data are stored in a database in MADCABS for use by MADCABS agents. For the case studies, the data are obtained from a simulator of the CSTR network, where multiple competing species coexist in the network and consume the same single resource. The ordinary differential equations modeling the operation of the reactors are written in C and connected to Repast Simphony through a Java Native Interface (JNI).

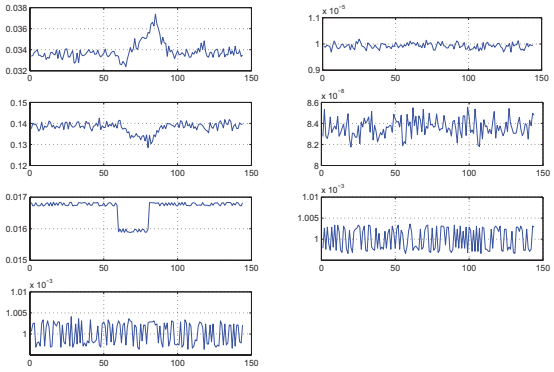


Fig. 4. A 5% process fault in the top right corner reactor 3.(a) Resource concentration in the reactor, (b) Species 1 concentration in the reactor, (c) Species 2 concentration in the reactor, (d) Species 3 concentration in the reactor (e) Feed flow rate into the reactor (Variable 5), (f) Outflowing interconnection to reactor 2 (g) Outflowing interconnection to reactor 7.

3.2 Monitoring and Fault Detection with MADCABS

The case studies use a four-by-five rectangular CSTR network, where three species coexist feeding from the same resource. Faults with different magnitudes and types are simulated to show the effectiveness of the agent-based monitoring, fault detection and diagnosis framework in MADCABS.

Fault Detection and Diagnosis. In Figure 4, a step decrease in the feed flow rate is introduced to reactor 3. The process fault affected the host species in the reactor since they start to die. The resource concentration in the reactor has increased after a delay after the dominant species start dying. From the figure, the variables that have been contributing to the fault are seen in the first three rows of the first column, the resource concentration in the reactor, dominant species concentration in the reactor and the feed flow rate to the reactor, which was reset to its original value after some time.

The contribution plots are given in Figure 5. For the five fault detection statistics that detected the existence of a fault in the system, PCA T^2, DPCA statistics and multiblock SPE showed that the main contributor to the fault is variable 5. PCA-SPE statistic had a smearing effect, where the signature of the fault could not be seen. Therefore, having multiple statistics improved diagnosis results as well. The fault has been detected on time by five fault detection agents.

Multiblock T^2 agent is insensitive to faults with magnitude less than 10%. A contribution chart that shows how many fault detection agents detected a contributing variable (Figure 6) indicates that variable 5, the feed flow rate to Reactor 3, is the common most contributor.

Table 2 provides a summary of 100 runs for each scenario. Multiblock PCA suffered from its insensitive T^2 agent, which had the highest missed alarm rate. The effect of the

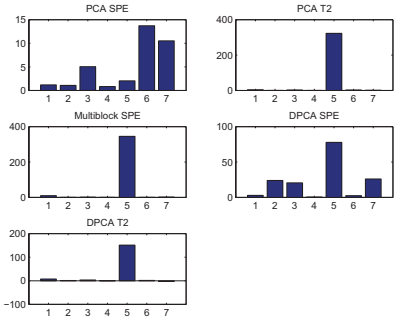


Fig. 5. Contribution plots for reactor 3 at the time of detection.

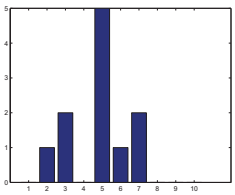


Fig. 6. Number of fault detection agents and common contributors for reactor 3.

insensitive statistic to the performance is realized in the performance of every combination with multiblock PCA. Combinations with DPCA improved the false and missed alarm rate. Especially PCA-DPCA performance is superior to any of the other less diverse combinations and seems to have a large impact on the combined performance when all three are used together. In summary, the results show that having multiple methods working together improves the effectiveness of the combined monitoring and fault detection.

Another type of fault in processes is sensor faults, where sensors might be defective and may provide false readings. A sensor fault should be identified in a timely manner since the measured variable can be used in computing the control actions and an erroneous reading may move the process to an undesired state, and may even destabilize the system. In general, correct and timely diagnosis and communication between control and diagnosis is required. A sensor fault in the form of a ramp decrease is given to the resource concentration sensor, variable 1 in reactor 6 (the figure is not provided because of space limitations). This sensor fault does not affect the other variables since it is not used for control.

The contribution plots show that the most contributing variable to the fault is variable 1. When this variable is taken out of the statistical model data, and a new model is built with one less variable, the new model reveals that the process is in-control. This indicates a potential sensor fault. The diagnosis results for reactor 6 are shared with control agents and also preprocessing agents in MADCABS so that preprocessing agents can provide reliable estimates instead of the faulty sensor, and control agents will continue to provide the necessary control actions to continue the desired operation level.

As another fault scenario, four consecutive faults are introduced to reactor 3. The fault is again introduced to the feed flow, and is of magnitude 1%. The fault introduction times and the detection times of the best performing combinations are given in Table 3. The reliability weight based consensus formation is shown to provide much earlier detection times than the majority based consensus criteria. Since the adapting reliability weight of the fault detection agents are taken into consideration, the first criteria provided earlier detection times, for consecutive faults. The results showed some kind of a learning pattern. However, this is going to be tested with different validation cases, where the training is followed by validation with different fault magnitudes. In Table 2, performance of DPCA-PCA combination was the same as the ALL combination, however, in Table 3, the detection times showed that when ALL of the monitoring agents are used in fault detection the fault is detected earlier than DPCA-PCA combination.

Table 3. Fault detection times (four consecutive process faults)

Agents	Fault#1	Fault#2	Fault#3	Fault#4
Actual Fault Times	220	250	290	330
PCA-DPCA (reliability)	221.9	250.9	290.4	330.8
ALL(reliability)	221.8	250.6	290.1	330.1
ALL(number)	223.1	254.4	294.0	333.6

The average number of agents that are flagging a fault when the consensus gives a fault flag is listed in Table 4 where two different consensus forming criteria are compared. When the agents' reliabilities increase with fault detection, less agents are required to declare a fault. When the presence of the majority of the fault detection agents is required to give a fault flag, the missed alarm rates increase and detection is delayed.

Table 4. Number of agents that flag fault at the time of detection (four consecutive process faults)

Agents	Fault#1	Fault#2	Fault#3	Fault#4
ALL(reliability)	4.06	3.46	3.48	3.25
ALL(number)	4.26	4.19	4.13	4.20

When the performances of different monitoring methods are compared the worst performing method is the multiblock PCA method because of the insensitive T^2 statistic. The overall performance of the PCA method is close to DPCA, but inferior because of the sensitive SPE statistic that gives many false alarms. The methods and the statistics are ranked and the results are provided in Table 5.

Table 5. Performances of the monitoring agents

Rank	Agent
1	DPCA SPE
2	Multiblock PCA SPE
3	PCA T^2
4	PCA SPE
5	DPCA T^2
6	Multiblock PCA T^2

4. SUMMARY AND CONCLUSIONS

PCA, DPCA and multiblock PCA methods are widely used multivariate SPM methods in process industries. However, all these methods are prone to false and missed alarms. The common practice in SPM is to test different monitoring tools form the literature, improve and tune the algorithms and find the best method that provides reliable monitoring. Considering the shortcomings of relying on a single SPM tool, consensus from several SPM tools is desirable. This is especially important for distributed and networked processes. Since there are multiple units, a monitoring system that is giving frequent false alarms on different operating units will be misleading and will not be relied on.

In order to improve the effectiveness of monitoring, several monitoring methods have been used together in the proposed framework. Some of the methods performed well on minor faults and disturbances, but had problems in contribution charts. Others gave good diagnostic results. Combining all these methods improved the effectiveness of the proposed overall monitoring, fault detection and diagnosis framework.

MADCABS provides an excellent environment to assess the performance of various SPM and fault detection methods for specific regions of process operation and adapt the reliance to different techniques based on prior experience and recursive assessment of performances. The agent management layer offers the tools and metrics to assess the performance of the monitoring, detection and diagnosis tools and dynamically update the confidence to specific techniques in a context-dependent way.

REFERENCES

Cinar, A., Palazoglu, A., and Kayihan, F. (2007). *Chemical Process Performance Evaluation.* CRC Press, Boca Raton, FL.

Jackson, J.E. (1980). Principal components and factor analysis: Part I-principal components. *J Qual Technol*, 12, 201–213.

Kourti, T. and MacGregor, J. (1996). Multivariate SPC methods for process and product monitoring. *J Qual Technol*, 28, 409–428.

Ku, W., Storer, R.H., and Georgakis, C. (1995). Disturbance detection and isolation by dynamic principal component analysis. *Chemometr Intell Lab*, 30, 179–196.

Qin, S., Valle, S., and Piovoso, M. (2001). On unifying mutliblock analysis with application to decentralized process monitoring. *J Chemometr*, 15, 715–742.

ROAD (2005). Repast organization for architecture and design. *Repast Simphony.* Available at http://repast.sourceforge.net.

Wangen, L. and Kowalski, B. (1988). A multiblock partial least squares algorithm for investigating complex chemical systems. *J Chemometr*, 3, 3–20.

Westerhuis, J., Kourti, T., and MacGregor, J. (1998). Analysis of multiblock and hierarchical PCA and PLS models. *J Chemometr*, 12, 301–321.

Wold, S., Kettaneh, N., and Tjessem, K. (1996). Hierarchical multiblock PLS and PC models for easier model interpretation and as an alternative to variable selection. *J Chemometr*, 10, 463–482.

Guaranteed Steady-State Bounds for Uncertain Chemical Processes

Jan Hasenauer [*], **Philipp Rumschinski** [**], **Steffen Waldherr**[*], **Steffen Borchers**[**], **Frank Allgöwer**[*], and **Rolf Findeisen**[**]

[*] *Institute for Systems Theory and Automatic Control, Universität Stuttgart, Germany (e-mail: {hasenauer,waldherr,allgower}@ist.uni-stuttgart.de)*

[**] *Institute for Automation Engineering, Otto-von-Guericke-Universität Magdeburg, Germany (e-mail: {philipp.rumschinski, steffen.borchers,rolf.findeisen}@ovgu.de)*

Abstract: Analysis and safety considerations of chemical and biological processes frequently require an outer approximation of the set of all feasible steady-states. Nonlinearities, uncertain parameters, and discrete variables complicate the calculation of guaranteed outer bounds. In this paper, the problem of outer-approximating the region of feasible steady-states, for processes described by uncertain nonlinear differential algebraic equations including discrete variables and discrete changes in the dynamics, is adressed.
The calculation of the outer bounding sets is based on a relaxed version of the corresponding feasibility problem. It uses the Lagrange dual problem to obtain certificates for regions in state space not containing steady-states. These infeasibility certificates can be computed efficiently by solving a semidefinite program, rendering the calculation of the outer bounding set computationally feasible. The derived method guarantees globally valid outer bounds for the steady-states of nonlinear processes described by differential equations. It allows to consider discrete variables, as well as switching system dynamics.
The method is exemplified by the analysis of a simple chemical reactor showing parametric uncertainties and large variability due to the appearance of bifurcations characterising the ignition and extinction of a reaction.

Keywords: Steady-states, nonlinear dynamical systems, discrete variables, hybrid dynamics, semidefinite programming, CSTR

1. INTRODUCTION

In the chemical and biochemical processing industry one frequently has to face large modelling uncertainties and process disturbances. Precise reaction mechanisms and kinetic parameters might be unknown and operating conditions, e.g. feed flowrate, or feed temperature, can be time dependent. Additionally, many of the substances handled in a chemical plant are potentially dangerous, e.g. inflammable or explosive. Reactions can lead to the disposal of large amounts of thermal energy, what makes safety considerations necessary. Stationary temperature and pressure have to stay below critical values and for instance in pharmaceutical processes the variability within the drug production has to be restricted. Hence, a detailed analysis of the process uncertainty is essential.

In this paper we address the problem of determining the set of all feasible steady-states of a process, for a class of uncertain hybrid nonlinear differential algebraic systems. Using the set of feasible steady-states the stationary process uncertainty can be upper bounded. Furthermore, it can be used to check whether for all possible disturbances, parameter variations, and operating conditions the process operates within previously defined constraints. One exemplary question to be asked is whether thermal runaway of a chemical reactor can be avoided under specific failure situations.

The physical processes taking place in chemical plants mostly behave in a continuous fashion. There are, however important discrete phenomena like changes in the physical system, e.g. phase transitions, imposed qualitative changes caused by limitation of the equipment, e.g. limited tank capacity, discontinuous input signals and process faults (Engell et al., 2000). To capture continuous as well as discrete phenomena, regime based approaches are used to model the process behavior (Seborg et al., 1989; Murray-Smith and Johansen, 1997; Lennartson et al., 1996). Frequently, one refers to this kind of models as hybrid models, because they contain both discrete and continuous dynamical components and an interface describing the interaction of them.

For most nonlinear systems an analytical calculation of the set of steady-states is impossible. Therefore, during the last decade several methods have been developed for approximating the set of feasible states, in the context of reachability analysis. Those methods are rather efficient if the considered system is linear time-invariant (Girard

and Guernic, 1996) and also for uncertain linear systems some results exist (Girard, 2005). However, if the systems under consideration are nonlinear, the approximation of the feasible set is more difficult. Asarin and coworkers developed an approach for two-dimensional systems based on piecewise linear approximation (Asarin et al., 2003) and Ramdani et al. (2008) proposed a method for high dimensional uncertain nonlinear systems using guaranteed set integration, which yields good results for cooperative systems. Nevertheless, the performance of these methods strongly depends on the particular structure of the nonlinear system and in many cases the results are very conservative.

Due to this drawback of set-based approaches, for the analysis of nonlinear systems, often simple Monte-Carlo type methods are employed (Robert and Casella, 2004). However, such approaches only provide the complete set of possible steady states in the limit of infinite many samples, i.e. important solutions might be left out, especially for highly nonlinear systems.

The method derived in this paper follows the idea presented in the work of Waldherr et al. (2008). There, recent advances in the field of semidefinite programming (SDP) (Parrilo, 2000; Chesi et al., 2003) are employed to compute certificates that a given set in state space cannot contain a steady-state for any feasible model parameterization. A very similar approach was earlier proposed by Kuepfer et al. (2007) for parameter estimation and later extended to dynamical systems by Borchers et al. (2009). However, all these methods are restricted to systems described by polynomial vector fields, which is rarely the case for chemical processes. Furthermore, discrete variables or parameters, as might occur in the analysis of chemical and biological processes, have not been considered.

In the following, an approach will be presented, which overcomes this shortcoming and allows the outer approximation of the set of all feasible steady-states of a process described by uncertain hybrid nonlinear differential algebraic equations with non-polynomial vector fields. Thus, systems combining continuous dynamics with logic or discrete components can be studied. Furthermore, a more elaborate algorithm is proposed to obtain a more precise approximation of the set of feasible steady-states, in cases the considered system has multiple steady-states.

The remainder of this paper is structured as follows: In Section 2 the problem of bounding the set of steady-states for processes described by non-polynomial hybrid differential algebraic equations is presented. Section 3 formalizes the problem statement. In Section 4 the resulting feasibility problem is relaxed to a semidefinite program which is used by the algorithm outlined in Section 5 to estimate the set of feasible steady-states. In Section 6 we provide as an example the analysis of a CSTR, before final conclusions are provided

Mathematical notation: The space of real symmetric $n \times n$ matrices is denoted as $\mathcal{S}^n$. N_a^b denotes the discrete set $\{1, \ldots, n_a^b\}$, where n_a^b is the number of considered variables. The positive semidefiniteness of a quadratic matrix $X \in \mathcal{S}^n$ is denoted $X \succeq 0$ and the trace of X by tr X. The transposed vector $(x^d)^T$ is written as x^{dT}.

2. PROBLEM STATEMENT

The processes under consideration are supposed to be described by hybrid differential algebraic systems which exhibit both continuous and discrete dynamical behavior. Such a process description is quite general. It covers for instance reaction networks which allow phase transitions, as well as discrete variables/inputs such as the opening of a valve or the on/off status of a heater. Mathematically, we assume that the process is described by

$$0 = F^d(\dot{x}^d, x^d, p^d, u^d), \qquad x^d(0) = x_0^d \tag{1}$$

Here $x^d \in \mathbb{R}^{n_x^d}$ is the state vector, $p^d \in \mathbb{R}^{n_p^d}$ the vector of parameters, $u \in \mathbb{R}^{n_u^d}$ the vector of inputs (externally manipulated variables), and $F^d : \mathbb{R}^{n_x^d} \times \mathbb{R}^{n_x^d} \times \mathbb{R}^{n_p^d} \times \mathbb{R}^{n_u^d} \to \mathbb{R}^{n_x^d}$ the mapping for a given discrete decision variable $d \in \mathbb{N}$. The decision variable d is assumed to be time dependent with $d(t) \in \mathcal{D}$.

To derive such hybrid differential algebraic descriptions in which each node captures the dynamics under certain operating conditions and to define switching surfaces is often easier than deriving ordinary differential equation models, holding for all process configurations.

In the following we are interested in the steady-state behavior of (1). The problem under consideration is to find all possible, or at least an outer bound of all, steady states of (1):

Problem 1. (Set of feasible steady states): Given the sets $\mathcal{D} \subset \mathbb{N}$, $\mathcal{P}^d \subset \mathbb{R}^{n_p^d}$ and $\mathcal{U}^d \subset \mathbb{R}^{n_u^d}$, compute the set $\mathcal{X}_s^*$ which contains all feasible steady-states of (1).

Note that the set of feasible steady-states for a given decision variable $d \in \mathcal{D}$ is defined by

$$0 = F^d(0, x^d, p^d, u^d). \tag{2}$$

Hence problem 1 can be split into n_d subproblems, where n_d is the cardinality of $\mathcal{D}$. For each subproblem one obtains a set of feasible steady states

$$\begin{aligned} \mathcal{X}_s^{d,*} = \{x^d \in \mathbb{R}^{n_x^d} \mid \exists\, p^d \in \mathcal{P}^d, u^d \in \mathcal{U}^d : \\ f^d(x^d, p^d, u^d) = 0\}, \end{aligned} \tag{3}$$

in which $f^d(x^d, p^d, u^d) = F^d(0, x^d, p^d, u^d)$. The whole set of feasible steady-states is given by the union of all steady-states

$$\mathcal{X}_s^* = \bigcup_{d \in \mathcal{D}} \mathcal{X}_s^{d,*}. \tag{4}$$

In the following the problem of computing an outer-approximation of $\mathcal{X}_s^*$ is considered. This was previously done by Waldherr et al. (2008) for differential equations with polynomial right hand sides. The main contribution of this paper is a generalization of these results to hybrid non-polynomial DAE systems.

3. BOUNDING BY PIECEWISE-POLYNOMIAL FUNCTIONS

The computational method we propose allows to handle uncertain systems that are described by polynomial equations. Therefore, (2) is transformed to a set of uncertain polynomial equations. In the case that f^d is rational, this can be trivially achieved by multiplying with the denominator. In cases in which the systems are non-rational, it is more difficult.

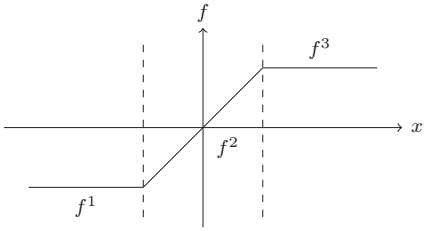


Fig. 1. Saturation function as example for the partitioning of piece-wise polynomial functions.

Savageau and Voit (1987) showed that any system with smooth non-polynomial nonlinearities can be converted to a polynomial system of larger state dimension, which is restricted via equality constraints to a manifold of the dimension of the original system. Unfortunately, in many cases the equality constraints are non-polynomial and so their method is not applicable for our approach. Instead, we apply a different method, which achieves a comparable result without enlarging the state space.

Piece-wise polynomial functions: In case that f^d is piece-wise polynomial, e.g. piece-wise linear, the state space can be partitioned into different intervals. This leads to an increase in the number of decision variables of the hybrid system and is illustrated in Figure 1 for the saturation function, which appears for instance if a process contains flow limiting valves. It has to be emphasized that in cases like this, the partitioning depends on the state. Thus, for a given region in state space $\mathcal{X}$ only a subset of decision variables $d \in \mathcal{D}$ is accessible.

General nonlinear functions: For functions which are not piece-wise polynomial, e.g. the exponential terms in the Arrhenius like rate constant, polynomial lower and upper bounds can be introduced as

$$\begin{aligned} g_1^d(x^d,p^d,u^d) \leq & f^d(x^d,p^d,u^d) \leq g_2^d(x^d,p^d,u^d) \\ & \forall x^d \in \mathcal{X}^d,\ p^d \in \mathcal{P}^d,\ u^d \in \mathcal{U}^d, \end{aligned} \tag{5}$$

in which $\mathcal{X}^d$ is the set in state space of interest. Using these bounds it can be shown that

$$\begin{aligned} \mathcal{X}_s^{d,*} \subseteq \{\ x^d \in \mathbb{R}^{n_x^d} | \exists\ p^d \in \mathcal{P}^d, u^d \in \mathcal{U}^d, c \in [0,1] : \\ c g_1^d(x^d,p^d,u^d) + (1-c) g_2^d(x^d,p^d,u^d) = 0 \big\}. \end{aligned} \tag{6}$$

Hence, the steady-state constraint $f^d(x^d,p^d,u^d) = 0$ can be substituted by the polynomial constraint

$$c g_1^d(x^d,p^d,u^d) + (1-c) g_2^d(x^d,p^d,u^d) = 0, \quad c \in [0,1], \tag{7}$$

where c has to be appended to p^d. This step corresponds to a constraint relaxation and $||f^d(x^d,p^d,u^d) - g_i^d(x^d,p^d,u^d)|| \ll 1$ should be enforced to keep the difference between $\mathcal{X}_s^{d,*}$ and the set of solutions of the relaxed problem small.

Combinations of the methods, e.g. rational, polynomial and nonlinear functions are possible, see Section 6.

4. BOUNDING STEADY STATES

In this section a method to compute an outer approximation of the state space region containing all steady-states is derived. For this purpose we define the feasibility problem,

$$(P): \begin{cases} \text{find} & d \in \mathcal{D},\ x^d \in \mathbb{R}^{n_x^d}, p^d \in \mathbb{R}^{n_p^d}, u^d \in \mathbb{R}^{n_u^d} \\ \text{subject to} & f^d(x^d,p^d,u^d) = 0 \\ & x^d \in \mathcal{X}^d,\ p^d \in \mathcal{P}^d,\ u^d \in \mathcal{U}^d, \end{cases}$$

which is in the following used for the classification of $\mathcal{X}^d$. If (P) is infeasible, $\mathcal{X}^d$ cannot contain any equilibrium points. (P) is called a mixed integer nonlinear program. Unfortunately, the feasibility problem (P) is in general non-convex and NP-hard.

Kuepfer et al. (2007) proposed a framework for relaxing a polynomial non-convex feasibility problem to a semidefinite program (SDP). Due to inherent convexity of SDPs, these problems can be solved computationally efficient, e.g. via primal-dual interior point methods. In the following, we present an approach which is based on the work of Kuepfer et al. (2007) and has been used for analysis of the set of feasible steady states in the case of biochemical reaction networks in Waldherr et al. (2008).

For the relaxation of (P) to a SDP, the original feasibility problem is at first rewritten as a quadratic feasibility problem (QP), for each d. Therefore, the vectors $\xi^d \in \mathbb{R}^{n_\xi^d}$ are introduced, which consists of the monomials of the model equation (1), i.e.

$$\xi^d = (1,\ x_i^d,\ p_j^d,\ u_k^d,\ x_i^d p_j^d,\ x_i^d u_k^d,\ p_i^d u_k^d, \ldots)^T \tag{8}$$

for all $i \in N_x^d$, $j \in N_p^d$, and $k \in N_u^d$. Using this monome vectors ξ^d, the equality constraints $f^d(x^d,p^d,u^d) = 0$ can be transformed to

$$0 = f_i^d(x^d,p^d,u^d) = \xi^{dT} Q_i^d \xi^d, \quad i \in N_x^d, \tag{9}$$

in which $Q_i \in \mathcal{S}^{n_\xi}$. Note that for higher order terms, additional constraints have to be introduced. For instance if ξ^d contains the second order term $x_1^d p_1^d$, the constraint $x_1^d p_1^d = x_1^d \cdot p_1^d$ must be introduced to express the dependency of the higer order monomial on the first order monomials. This leads to additional constraints of the form,

$$\xi^{dT} Q_i^d \xi^d = 0, \quad i \in N_c^d, \tag{10}$$

in which $Q_i \in \mathcal{S}^{n_\xi}$, $N_c^d = \{n_x^d + 1, \ldots, n_x^d + n_c^d\}$, and n_c^d is the number of dependencies. To simplify the notation we set $N_{xc}^d = N_x^d \cup N_c^d$.

To further simplify the notation we restrict $\mathcal{X}^d$, $\mathcal{P}^d$, and $\mathcal{U}^d$ to be generated by the intersection of half-spaces, e.g. $\mathcal{X}^d$, $\mathcal{P}^d$, and $\mathcal{U}^d$ can be convex polytopes. In this case, $x^d \in \mathcal{X}^d$, $p^d \in \mathcal{P}^d$, and $u^d \in \mathcal{U}^d$ can be written as

$$B^d \xi^d \geq 0, \tag{11}$$

in which $B^d \in \mathbb{R}^{n_b^d \times n_\xi^d}$, and n_b^d is the sum of constraints on x^d, p^d, and u^d.

The original feasibility problem (P) can then be restated as

$$(QP): \begin{cases} \text{find} & \xi^d \in \mathbb{R}^{n_\xi^d},\ d \in \mathcal{D} \\ \text{subject to} & \xi^{dT} Q_i^d \xi^d = 0,\ i \in N_{xc}^d \\ & B^d \xi^d \geq 0 \\ & \xi_1^d = 1. \end{cases}$$

Using the ideas suggested by Parrilo (2003), the (QP) is subsequently relaxed to a SDP, for each d, by introducing the matrices $X^d = \xi^d \xi^{dT}$ and dropping the appearing non-convex constraint $\text{rank}(X^d) = 1$. This leads to the relaxed feasibility problem

$$(RP): \begin{cases} \text{find} & X^d \in \mathcal{S}^{n_\xi^d},\ d \in \mathcal{D} \\ \text{subject to} & \operatorname{tr}(Q_i^d X^d) = 0,\ i \in N_{xc}^d \\ & B^d X^d e_1^d \geq 0 \\ & B^d X^d B^{dT} \geq 0 \\ & \operatorname{tr}(e_1^d e_1^{dT} X^d) = 1 \\ & X^d \succeq 0, \end{cases}$$

in which $e_1^d = (1, 0, \ldots, 0)^T \in \mathbb{R}^{n_\xi^d}$. Note that the relaxation may induce additional solutions. To reduce conservatism, the redundant constraint $B^d X B^{dT} \geq 0$ is added, which is fulfilled by every solution of the problem (QP) (Kuepfer et al., 2007).

From (RP) one can derive the Lagrange dual problem (DP_d) for each d,

$$(DP_d): \begin{cases} \text{maximize} & \nu_1^d \\ \text{subject to} & e_1^d \lambda_1^{dT} B^d + B^{dT} \lambda_1^d e_1^{dT} + B^{dT} \lambda_2^d B^d \\ & \quad + \lambda_3^d + \nu_1^d e_1^d e_1^{dT} + \sum_{i \in N_{xc}^d} \nu_{2,i}^d Q_i^d = 0 \\ & \lambda_1^d \geq 0,\ \lambda_2^d \geq 0,\ \lambda_3^d \succeq 0, \end{cases}$$

in which the Lagrange multipliers are $\lambda_1^d \in \mathbb{R}^{n_b^d}$, $\lambda_2^d \in \mathcal{S}^{n_b^d}$, $\lambda_3^d \in \mathcal{S}^{n_\xi^d}$, $\nu_1^d \in \mathbb{R}$ and $\nu_2^d \in \mathbb{R}^{n_x^d + n_c^d}$ (Waldherr et al., 2008). Using the dual problem, one can obtain an infeasibility certificate for the original problem.

Lemma 2. Let $\nu_1^{d,*}$ be the optimal cost of (DP_d). If

$$\inf \left\{ \nu_1^{d,*} \mid d \in \mathcal{D} \right\} = \infty, \tag{12}$$

then the original feasibility problem (P) is infeasible.

This follows directly from weak duality. Only if the Lagrange dual problem is unbounded from above for all $d \in \mathcal{D}$ the infeasibility of (P) can be guaranteed. The advantage of the formulation using the Lagrange duals is that all subproblems are convex and can be solved efficiently.

In case that $\operatorname{card}(\mathcal{D}) \gg 1$, checking all the distinct combinations of decision variables can become very costly. One possibility to reduce the problem size is to divide $\mathcal{D}$ into subsets $\mathcal{D}_i$. The subsets $\mathcal{D}_i$ can be merged to a common node and the analysis can be performed for all subsets instead of for all nodes. This approach can also be combined with a hierachical refinement of the subsets $\mathcal{D}_i$, which reduces the computational demand significantly.

5. ALGORITHM

Using the Lagrange dual problem (DP_d), certificates for the infeasibility of (4) can be computed. This allows to exploit (DP_d) to determine an outer approximation $\mathcal{X}_s$ of $\mathcal{X}_s^*$. In this work, this is done via simple a multidimensional bisection algorithm (Jaulin et al., 2001). Compared to the work by Waldherr et al. (2008) this allows a better approximation of $\mathcal{X}_s^*$ but is computationally more demanding. The basic implementation can be summarized as follows:

Algorithm: $\mathcal{X}_s = \text{Approximation-}\mathcal{X}_s^*(\mathcal{X}, \mathcal{P}, \mathcal{D})$

1. If volume$(\mathcal{X}) < \epsilon$, return $\mathcal{X}_s = \mathcal{X}$
2. Check feasibility of $DP_d(\mathcal{X}, \mathcal{P}, \mathcal{D})$, $\forall d \in \mathcal{D}$
3. If $\inf \left\{ \nu_1^{d,*} \mid d \in \mathcal{D} \right\} = \infty$, return $\mathcal{X}_s = \emptyset$
4. If $\inf \left\{ \nu_1^{d,*} \mid d \in \mathcal{D} \right\} \neq \infty$:
   - 4.1. Bisection of $\mathcal{X}$ in $\mathcal{X}_1$ and $\mathcal{X}_2$
   - 4.2. $\mathcal{X}_{1,s} = \text{Approximation-}\mathcal{X}_s^*(\mathcal{X}_1, \mathcal{P}, \mathcal{D})$
   - 4.3. $\mathcal{X}_{2,s} = \text{Approximation-}\mathcal{X}_s^*(\mathcal{X}_2, \mathcal{P}, \mathcal{D})$
   - 4.4. Return $\mathcal{X}_s = \mathcal{X}_{1,s} \cup \mathcal{X}_{2,s}$

Remark 3. Note that for the application of this algorithm an initial set $\mathcal{X}_0$ must be chosen. If we want to guarantee that an outer approximation of $\mathcal{X}_s^*$ is found containing all feasible equilillbrium points, $\mathcal{X}_s^* \subseteq \mathcal{X}_0$ must hold. This is not a restriction because a suitable $\mathcal{X}_0$ can often easily be determined from physical insight into the problem.

6. BOUNDING THE STEADY STATES OF A CSTR

In order to illustrate the proposed scheme the steady-state behavior of a CSTR is analyzed. The reactor considered is a simple tank filled with fluid stirred by an impeller, an inflow and an outflow, as depicted in Figure 2.

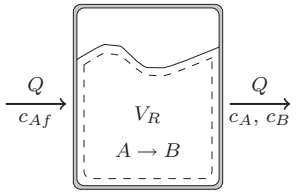


Fig. 2. Schematic of the considered simple CSTR.

6.1 System description

Specifically we consider an adiabatic, constant volume CSTR in which the first-order, exothermal liquid-phase reaction

$$A \xrightarrow{k} B$$

takes place. The conversion rate is given by $R = k(T)c_A$, in which the reaction rate constant is modelled using Arrhenius' equation,

$$k(T) = k_\infty e^{-\frac{E}{RT}}. \tag{13}$$

Simple mass and energy balances lead to the following set of ordinary differential equations:

$$\begin{aligned} \frac{dc_A}{dt} &= \frac{1}{\theta}(c_{Af} - c_A) + k(T)c_A \\ \frac{dT}{dt} &= \frac{1}{\theta}(T_f - T) - \frac{\Delta H_R}{C_p \rho} k(T) c_A, \end{aligned} \tag{14}$$

which captures the dynamics of the CSTR (Rawlings and Ekerdt, 2002). The state variables are the concentration c_A of reactant A, and the reactor temperature T. The parameters are the mean residence time $\theta = V_R/Q$, the reactor volume V_R, the flowrate Q, the concentration of A in the feed stream c_{Af}, the feed stream temperature T_f, the reaction enthalpy ΔH_R, the heat capacity of the fluid C_p, and the fluid density ρ. The numerical values of the nominal parameters are provided in Table 1.

6.2 Analysis of the nominal CSTR

In case that all parameters are known, one can exactly predict how the reactor behaves in different operating conditions. Hereby, since the mean residence time θ is the easiest parameter to manipulate, the operating condition will be defined in terms of θ. The other parameters are assumed to be fixed.

Table 1. Parameter values.

Parameter	Value	Units	Uncertainty
T_f	298	K	3K
C_p	4.0	KJ/kg K	5%
c_{Af}	2.0	kmol/m^3	5%
k_∞	5.0×10^8	min^{-1}	5%
E/R	8.0×10^3	K	–
ρ	10^3	kg/m^3	–
ΔH_R	-3.0×10^5	kJ/kmol	5%

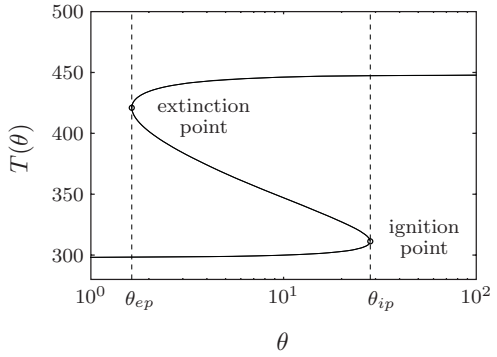


Fig. 3. Bifurcation diagram of CSTR without parameter uncertainties.

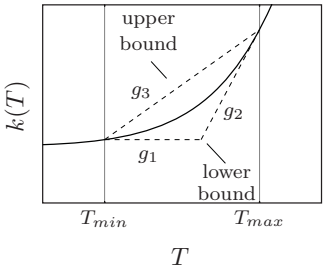


Fig. 4. Bounding of $(-)$ Arrenhius term with $(--)$ linear functions.

Using continuation methods it is possible to numerically compute the steady-state curve (bifuraction diagramm) for varying residence times (Dhooge et al., 2003), as shown in Figure 3. θ_{ep} and θ_{ip} denote the mean residence time at the extinction and the ignition point respectively.

6.3 Analysis of CSTR with parameter uncertainties

If one or more parameters are uncertain, which is in practice always the case, calculating the set of steady-state is significantly more challenging. Typically, sampling based techniques such as Monte-Carlo like methods are used. These allow the approximation of the union of all feasible equilibrium points $\mathcal{X}_s^*$. However, as for all Monte-Carlo like methods no bounds for the obtained sets can be provided. Our approach overcomes this problem and enables us to compute an outer approximation of the set of feasible equilibrium points of the uncertain system.

Approximation of the rate constant: Applying the proposed method requires in a first step to bound the Arrhenius-like rate constant from below and from above using polynomial functions. In this paper $k(T)$ is bounded via three linear functions,

$$\max(g_1, g_2) \leq k \leq g_3, \quad \forall T \in [T_{min}, T_{max}], \tag{15}$$

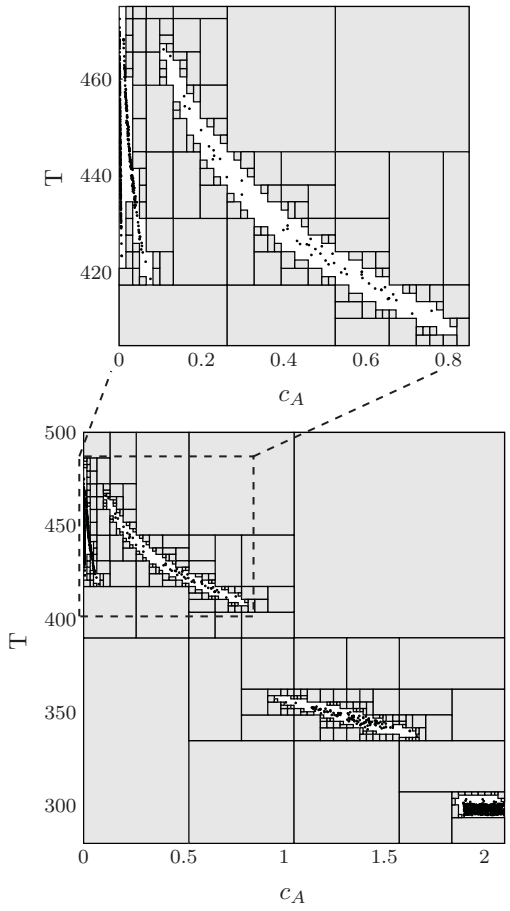


Fig. 5. Region in state space which cannot contain steady-states for given parameter uncertainties and $\theta \in \{1, 10, 100\}$ versus $(\cdot)$ steady-states computed using Monte-Carlo sampling.

as depicted in Figure 4. This approach is very simple and has the disadvantage that the approximation of $k(T)$ is less precise if the difference of T_{min} and T_{max} becomes large. Therefore, we don't use a static approximation but rather select g_1, g_2 and g_3 in each interation of the bisection algorithm dependent on the box $\mathcal{X}$ in state space currently under consideration. This allows to keep the overestimation of the set of feasible steady-states small as will be seen later.

One could of course choose other methods to bound $k(T)$, for instance based on high order polynomials and the Taylor series expansion, but in many cases the computational effort to solve the semidefinite program once will increase significantly and the presented simplistic approach will be more efficient.

Set of feasible steady-states: The above derived theory and the bounding of $k(T)$ allow to compute the set of feasible steady-states of the CSTR. As decision variable we consider besides the temperature interval also the mean residence time θ. Additionally, most parameters are uncertain. The amounts of uncertainty with respect to the nominal values are provided in Table 1.

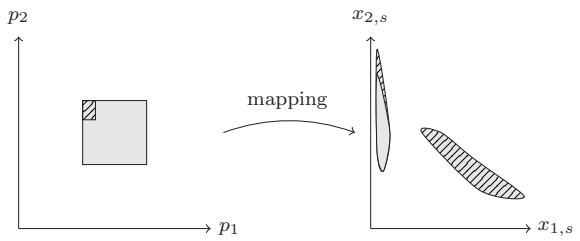


Fig. 6. Illustration of the nonlinear mapping from parameter to steady-states.

The algorithm outlined in Section 5 is in the following used to estimate the set of all feasible equilibrium points of (14) for the given parameter uncertainties and $\theta \in \{1, 10, 100\}$. The results are shown in Figure 5, where the part of the state space which is certified infeasible is marked light gray. To compare our results with classical approaches, five thousand equally distributed Monte-Carlo samples for the accessible parameter set were taken and the steady-states were determined.

Computation of the set of feasible steady-states: As one can see, the results match with each other. However, a closer look at the results reveals several disadvantages of the sampling based approach. First of all, the number of samples in some regions of the state space is small compared to other regions, where the sampling density is extremely high. This indicates that many parameters lead to steady-states in the region with high sampling density, but there are still some regions that cannot be explored unless even higher numbers of samples are used. This might represent a problem, whenever the set of all feasible steady-states has to be computed, since normally a homogeneous sampling rate is more desireable. However, the Monte-Carlo method is not able to guarantee under such a condition that the whole state space is explored, due to the highly nonlinear mapping between parameters and steady-states, illustrated in Figure 6. Therefore, the set of feasible equilibrium points is always underestimated, even for exhaustive Monte-Carlo sampling, while the proposed method guarantees that all equilibrium points are contained in the determined set.

7. CONCLUSION

In this work we studied the problem of outer bounding the region in state space containing all equilibrium points of uncertain hybrid differential algebraic systems. The proposed method is based on the formulation as a feasibility problem and a relaxation to a SDP. It is shown that guaranteed outer bounds of the feasible set of equilibrium points can be determined.

The advantage of the proposed methodology in comparison to Monte-Carlo based approaches is explained and shown considering a simple CSTR process. In particular, the developed method does not rely on sampling and can deal with strongly nonlinear and non-unique mappings from parameters to steady-states.

The computed set is guaranteed to contain all feasible steady-states, thus worst case scenarios can be analyzed. This is of certain interest to evaluate controller performance in fault situations.

8. ACKNOWLEDGEMENTS

This work was supported by the Forschungseinheiten der Systembiologie (FORSYS) [grant 0315280D]; the International Max Planck Research School Magdeburg; and by the Stuttgart Research Centre for Simulation Technology.

REFERENCES

Asarin, E., Dang, T., and Girard, A. (2003). Reachability analysis of non-linear systems using conservative approximations. *Hybrid Systems: Computation and Control*, 2623, 20–35.

Borchers, S., Rumschinski, P., Bosio, S., Weismantel, R., and Findeisen, R. (2009). Model invalidation and system identification of biochemical reaction networks. *Proceedings of the 16th IFAC Symposium on Identification and System Parameter Estimation (SYSID 2009)*. To appear.

Chesi, G., Garulli, A., Tesi, A., and Vicino, A. (2003). Characterizing the solution set of polynomial systems in terms of homogeneous forms: an lmi approach. *International Journal of Robust and Nonlinear Control*, 13(13), 1239–1257.

Dhooge, A., Govaerts, W., and Kuznetsov, Y. (2003). Matcont: A Matlab package for numerical bifurcation analysis of odes. *ACM Transactions on Mathematical Software*, 29(2), 141–164.

Engell, S., Kowalewski, S., Schulz, C., and Stursberg, O. (2000). Conitinuous-discrete interactions in chemical processing plants. *Proceedings of the IEEE*, 88(7), 1050–1068.

Girard, A. (2005). Reachability of uncertain linear systems using zonotopes. *Hybrid Systems: Computation and Control*, 3414, 291–305.

Girard, A. and Guernic, C.L. (1996). Zonotope/hyperplane intersection for hybrid systems reachability analysis. *IEEE control systems*, 16(5), 45–56.

Jaulin, L., Kieffer, M., Didrit, O., and Walter, E. (2001). *Applied interval analysis*. Springer, Heidelberg.

Kuepfer, L., Sauer, U., and Parrilo, P. (2007). Efficient classification of complete parameter regions based on semidefinite programming. *BMC Bioinformatics*, 8, 12.

Lennartson, B., Tittus, M., Ehardt, B., and Pettersson, S. (1996). Hybrid systems in process control. *IEEE control systems*, 16(5), 45–56.

Murray-Smith, R. and Johansen, T.A. (1997). *Multiple Model Approaches to Modelling and Control*. Taylor and Francis, London.

Parrilo, P. (2000). *Structured Semidefinite Programs and Semialgebraic Geometry Methods in Robustness and Optimization*. Ph.D. thesis, Caltech, Pasadena, CA.

Parrilo, P. (2003). Semidefinite programming relaxations for semialgebraic problems. *Math. Program., Ser. B*, 96, 293–320.

Ramdani, N., Meslem, N., and Candau, Y. (2008). Reachability analysis of uncertain nonlinear systems using guaranteed set integration. *Proceedings of the 17th IFAC World Congress*, 8972–8977.

Rawlings, J. and Ekerdt, J. (2002). *Chemical reactor analysis and design fundamentals*. Nob Hill Publishing, Madison, WI 53705.

Robert, C.P. and Casella, G. (2004). *Monte Carlo Statistical Methods*. Springer-Verlag.

Savageau, M.A. and Voit, E.O. (1987). Recasting nonlinear differential equations as S-systems: a canonical nonlinear form. *Mathematical Biosciences*, 87, 83–115.

Seborg, D., Edgar, T., and Mellichamp, D. (1989). *Process Dynamics and Control*. Wiley.

Waldherr, S., Findeisen, R., and Allgöwer, F. (2008). Global sensitivity analysis of biochemical reaction networks via semidefinite programming. *Proceedings of the 17th IFAC World Congress*, 9701–9706.

Extremely Fast Catalyst Temperature Pulsing: Design of a Prototype Reactor

Jasper Stolte * **Ton Backx** * **Okko Bosgra** *

* *Eindhoven University of Technology, PO Box 513, 5600 MB Eindhoven, the Netherlands (e-mail: j.stolte@tue.nl).*

Abstract: This paper discusses a novel principle of advanced process control strategy: the extremely fast and local pulsing of temperature. This strategy leads to some interesting potential applications, but there are no devices implementing it available yet. One such device currently under construction by the authors is introduced in this paper. It operates by converting electrical energy into heat by forcing a very high current through a very thin resistive element, which also acts as the catalyst for heterogeneous reactions. A design procedure for the key parameters is developed and a simulation of heat distribution in the design under construction is presented. The simulation shows that it should be possible to get local temperature peaks of 500 K which exist for only about 20 μs.

Keywords: Process Control, Pulses, Heterogeneous Catalysis, Temperature Forcing, Periodic Control, Non-Steady State.

1. INTRODUCTION

For various reasons, chemical reaction engineers prefer to operate reactions in steady state. In modeling and design of chemical processes, reactors are almost always assumed to be ideally mixed. Control of such processes boils down to maintaining constant reaction conditions as best as possible. However, chemical reaction systems can be seen as systems of highly nonlinear partial differential equations, which are typically full of transient dynamics. If we know that these dynamics can be present, is it not possible to find cases where the dynamics can be exploited?

Attempting to exploit the dynamics demonstrated by chemical processes is not new. Horn and Lin (1967) already introduces periodic reactor operation, and it is shown there that there are reaction complexes for which periodic operation is fundamentally better than any steady state operation. This idea was picked up by researchers all over the world, and in the following decades there have been numerous studies showing there exist reaction complexes for which non steady state operation gives fundamental advantages, see Bailey (1973); Matsubara et al. (1973); Silveston et al. (1995); Silveston and Hudgins (2004) for excellent reviews of this research field.

Although there are plenty examples of academic (often theoretic) research, there are few known examples of practical applications in industry. One promising direction of research which has already shown significant improvements in a number of studies is microwave assisted chemistry, sometimes called microwave enhanced chemistry Wan et al. (1990); Wan (1993); Will et al. (2004). When some parts of a reactor subjected to microwave irradiation are more susceptible than others, they will heat up more and create gradients in temperature. Wan (1993) specifically considers heating the catalyst particles directly. Microwave assisted catalysis is not very accessible to most scientists, as powerful microwave equipment is expensive, it is difficult to model due to inhomogeneities, and measurements are difficult also. Still, microwave heating has shown to allow modes of operation that are impossible using steady state operation.

Silveston and Hudgins (2004) state that the field of temperature forcing is relatively underdeveloped compared to the periodic feed forcing, even though the reaction complexes show more strong non-linear behavior as a function of temperature. The reason given is that due to the large time constants and energy flows involved it is difficult to apply temperature forcing to large amounts of matter. That paper also indicates that the advance of micro-reactors may change allow for better heat transfer paving the way for new studies in the forced temperature direction.

The authors of this paper have started a project on dynamic operation of heterogeneous catalysis. A first theoretical result showed that very fast and local pulsing of temperature may in some cases fundamentally improve attainable results, see Stolte et al. (2008). Also, Wan et al. (1990) state that the fastest and most intense pulses of microwave energy shows the most interesting results for the important methane coupling reaction complex. Work has started on a setup that will create very fast and intense temperature pulses, of several hundred degrees Kelvin in tens of microseconds. To our knowledge this is orders of magnitudes faster than any setups reported in literature have realized.

This paper considers the considerations in designing a setup for heterogeneous catalysis capable of such temperature pulses, delivered directly at the catalyst. Section 2 describes the general reactor setup, as well as a design procedure to decide upon the critical design parameters. Section 3 considers a simulation of the heat distribution

within the reactor, which suggests that heating and cooling within such short timescales is possible. Section 4 discusses a few potential pitfalls that may be encountered when designing such a reactor.

2. REACTOR CONCEPT

The goal of this setup is to create a reactor which is capable of operating heterogeneous catalysis reactions where the catalyst is turned into an actuator which creates very fast and large pulses in temperature. Pulse shaped temperature profiles indicate that very fast heating as well as very fast cooling are required. A very schematic 2D-view of the reactor under construction is given in figure 1.

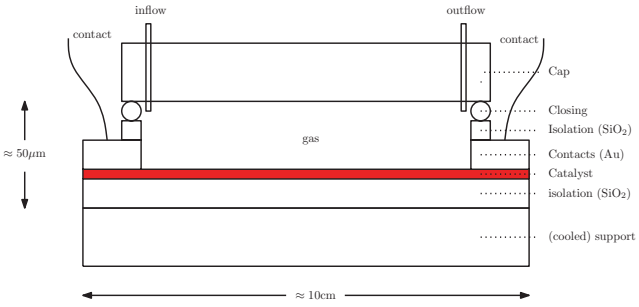


Fig. 1. Schematic view of the prototype reactor concept (not to scale). A channel is created where gas can flow through, with a floor consisting of a thin catalyst layer which is heated.

The concept is based upon a micro-flow-channel reactor, but it has a 'floor' consisting of a thin layer of catalyst through which an electrical current can be driven. Contacts are mounted at the sides to connect a source of electrical energy. For this general heterogeneous catalysis setup the catalyst is chosen to be platinum, which has excellent conductive properties. The reactor heating is supplied by the electrical source, but the cooling cannot be done actively. The heat simply has to flow to cooler areas to cool down the catalyst layer. The bulk of the energy in the catalytic layer will be conducted through the SiO_2 layer to the the cooled support which consists of pure silicon. To get a nice pulse like temperature profile it is desired to have the time constant for cooling approximately equal to the time constant for heating. If it is too small, the energy will flow away already while the pulse is still being applied, and if it is too large it will simply take a long time to cool down. The thickness of especially the catalyst layer and the SiO_2 layer supporting it are critical degrees of freedom to shape the time constant for cooling.

Reasons to choose for this design are:

- Energy is by definition directly added to the catalyst
- Fast creation of electrical currents is a well developed field
- Platinum resistance dependends on temperature almost linearly. By measuring voltage over and current through the catalyst layer an indication for its temperature can be found.
- By electrochemical deposition very thin layers of other catalyst can be deposited on the platinum, giving flexibility in using different catalysts.
- There are no fundamental reasons preventing this design from being scaled up for application in industry.

2.1 Basics of Electrical Heating

This subsection will summarize the basics of electrical heating as needed for this application. When voltage and current are used to heat a layer of resistive material and no heat is lost to the environment, all electrical energy is directly converted into temperature and the following relation holds:

$$\frac{dT}{dt} = \frac{P}{\rho c_p V} \tag{1}$$

where T is temperature [K], P is electrical power [W], ρ is material density [kg/m^3], c_p is specific heat [J/kg K] and V is volume [m^3]. Since the layer is rectangular in shape it can be described by a certain length l [m], height h [m] and width w [m], which together make up the layer volume as given in (2). Also there are the basic electrical relations for power and resistance given in (3) and (4):

$$V = lhw \tag{2}$$

$$P = \frac{U^2}{R} \tag{3}$$

$$R = \frac{l}{\sigma w h} \tag{4}$$

where R is the electrical resistance [Ω] and σ is the electrical conductivity [S/m]. Combining all the equations above, the following relation is found for the temperature gradient due to electrical heating:

$$\frac{dT}{dt} = \frac{\sigma}{\rho c_p}\frac{U^2}{l^2} \tag{5}$$

The relation as stated in (5) is used in the simulation of section 3. It can be seen from (5) that the heating is dependent on some material parameters, and on the applied voltage squared per unit length squared. This means that if a higher temperature gradient is desired for a given setup the voltage should be increased, or the length should be decreased.

The limiting factor is the electrical current. The voltage in the formulas above can only exist if the corresponding current runs through the material. The electrical current I [A] for a voltage applied to a layer of material is given by Ohms law:

$$I = \frac{U}{R} \tag{6}$$

From this relation it is evident that by increasing the voltage, the current increases also. Furthermore making the length smaller decreases the resistance according to (4) and therefore will also increase the current. Although there is no fundamental limit to the current, there is a practical limit in how quickly a large current can be created and switched.

2.2 Resonant Circuit for Energy Transfer

To create short bursts of energy, a resonant circuit is used. The basic resonant circuit is shown in figure 2, where R represents the resistance of the catalyst layer. This layer is connected to two external components, a capacitor C and an inductance L. The capacitor is charged to a certain

voltage separately, and then connected to the other two components resulting in the circuit of figure 2. Due to the charge in the capacitor, current will flow and through the inductance and the resistance dissipating energy which is released as heat.

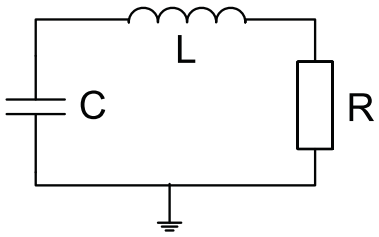


Fig. 2. Basic resonant circuit consisting of a capacitance, an inductance and a resistance. The resistance represents the catalytic layer.

From linear circuit theory, the circuit of figure 2 is governed by (7):

$$U_R = \frac{sRC}{s^2LC + sRC + 1} U_{C0} \tag{7}$$

where U_R is the voltage over the catalytic layer [V], U_{C0} is the initial voltage over the capacitor [V] and s is the Laplace operator. This equation shows second order dynamics in the numerator, which can be associated with a natural frequency ω_0 [rad/s] and a quality factor Q [-] as given in (8).

$$s^2/\omega_0^2 + s/\omega_0 Q + 1 = s^2LC + sRC + 1 \tag{8}$$

The natural frequency gives the frequency of the oscillation in the circuit, and the quality factor roughly indicates how many periods can be seen before the oscillation is gone and all the energy from the capacitor is dissipated in the resistance. From (8) the natural frequency and quality factor for a given circuit can be calculated:

$$\omega_0 = \frac{1}{\sqrt{LC}} \tag{9}$$

$$Q = \frac{1}{R}\sqrt{\frac{L}{C}} \tag{10}$$

In this application it is desired to release the energy stored in the capacitor as quickly as possible in the catalyst layer. A quality factor of 0.5 together with a high natural frequency is considered optimal.

2.3 Design Strategy

The question of how to choose values for R,L,C etcetera is an important one. This subsection proposes a strategy for consistent selection of most of the free parameters. The simulation introduced in section 3 is needed to verify the time constant for cooling and verifying validity of the thickness of the catalyst layer. The following design strategy is proposed:

(1) Fix resistance R and its dimensions: The value for the resistance should be the dominant resistance in the circuit. Wires and connections can easily account for up to 1 Ω, and the catalyst layer should have significantly more resistance. A value of 100 Ω is chosen. For this laboratory setup it is desired that the amount of energy needed to create a significant heat pulse is not too large, so volume should be kept small. At the same time it is desired that the width and length of the layer are much larger than the height, such that the energy loss to the sides can be neglected. Also the sizes should be chosen such that the device can be created using integrated circuit techniques. The layer should be very thin, a value of 100 nm is chosen for the height h. To make the resistance approximately 100 Ω, we choose w to be 0.5 mm and l to be 5 cm.

(2) The next step is to determine the pulse amplitude. Since we want to significantly influence kinetics for a very short time, the temperature pulse should be large. A value of 1000 K is chosen. Since the dimensions are already specified, the amount of energy needed to make the platinum layer temperature rise by this amount can easily be calculated. For the selected dimensions, this energy is 7.1 mJ. This value holds under the assumption that no energy is lost during the application of the pulse, which in practice will be the case. In the next section it is shown that for this design the simulation predicts an actual rise of 500 K which the authors find satisfactory.

(3) Determine the maximum available voltage. The voltage is limited mostly by the switching devices needed to open and close the current loop. In this design a MOSFET type device will be used for switching. Even high voltage MOSFETs can typically not deal with voltages greater than 1000 V. This will be the voltage used. For higher voltages spark gap switches can be applied if necessary. The voltage determines the size of the capacitance C needed by the basic relation of (11).

$$E = \frac{1}{2} C U_{C0}^2 \tag{11}$$

In this case the capacitance needed is 14 nF. Using this capacitance value, the corresponding inductance can be computed from (10) to get the proper quality factor for the resonance circuit. The natural frequency that corresponds to this network can be computed using (9), in this case it is 1.32 Mrad/s. That means the period length is approximately 4.75 μs, which is excellent for this application. If the natural frequency would have been too low, the applied voltage would have to be increased further.

Table 1. Design Parameters Chosen

Description	Parameter	Value	Unit
Resistance	R	106	[Ω]
Catalyst length	l	50	[mm]
Catalyst width	w	0.5	[mm]
Catalyst height	h	100	[nm]
Temperature rise	ΔT	1000 (500)	[K]
Pulse energy	E	7.1	[mJ]
Quality factor	Q	0.5	[−]
Capacitance	C	14	[nF]
Inductance	L	40	[μH]
Natural Frequency	ω_0	1.32	[Mrad/s]

Table 1 summarizes the choices made using this design procedure.

3. HEAT SIMULATION MODEL

The previous section introduced the design of a pulsed reactor, and in this section the heat distribution within that design is simulated. Since the design is developed in such a way that heat loss to the sides should be negligible in comparison to the heat loss to the support of the catalytic layer, only the vertical dimension is simulated. Figure 3 schematically shows the layers (not drawn to scale), with their respective sizes in one dimension which is named the x dimension.

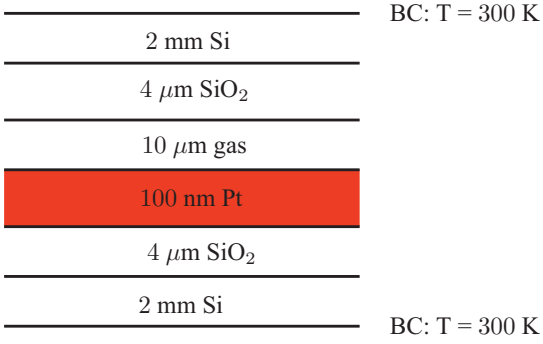


Fig. 3. Schematic view of the layers in the vertical dimension. In this dimension the heat distribution will be simulated. BC stands for boundary condition.

Heat distribution by conduction and diffusion in the model is simulated, as a lower limit to the cooling. The partial differential equation that is being solved is the standard heat diffusion equation given as (12), where κ is the local thermal conductivity [m^2/s].

$$\frac{dT}{dt} = \kappa(x)\frac{\partial^2 T}{\partial x^2} \tag{12}$$

Since this is a very straightforward geometry, the pseudo spectral method is used for collocation (Trefethen (2000); Weideman and Reddy (2000)) to get the benefit of spectral accuracy. Each of the layers is collocated separately, with energy preserving von Neumann boundary conditions at the layer boundaries. Only at the very top and bottom there are fixed temperatures (Dirichlet boundary conditions). Initially, all the layers have a temperature of 300 K. When heat enters the system in the catalytic layer, conduction will transport the heat through the other layers to the outer boundary where it is eventually lost to the surroundings.

After collocation, a system of ordinary differential equations which pose an initial value problem that is solved in Matlab using the built-in stiff ODE solvers.

3.1 Simulation Results

An RLC circuit is also simulated in parallel with the heating simulation. The RLC parameters are chosen identical to those find using the design procedure described in the previous section. The capacitor discharges via the inductance into the catalytic layer. The power that is released as heat into the catalytic layer is shown in figure 4. The peak power is about 5 kW which creates an enormous temperature gradient in the thin catalyst layer.

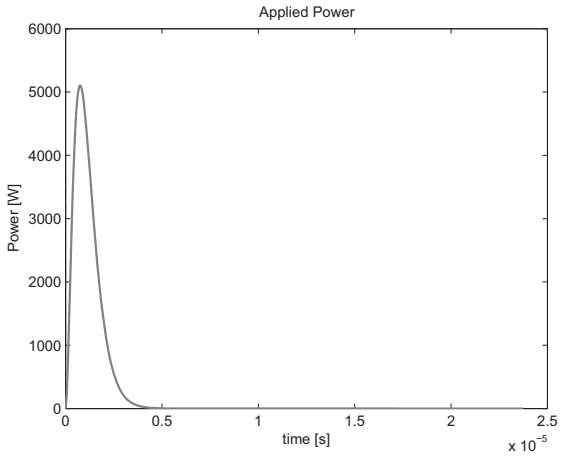


Fig. 4. Power supplied to the catalytic layer by the resonant circuit. The shape is very much like a pulse, with the peak power at about 5 kW.

This heat will result in a temperature rise of the catalytic layer, which will therefore have a temperature gradient with respect to the other layers. Transport of heat to the other layers will result. The temperature at the boundary between the catalytic layer and the gas layer is the surface temperature, which is critical for the catalytic reactions taking place. The simulated surface temperature resulting from the energy pulse is shown in figure 5. The surface temperature rises by about 500 K and not with the 1000 K for which this amount of energy (7.1 mJ) was computed in the previous section. This is due to the loss of energy to the other layers while the pulse is still in progress. About half the energy is lost to the other layers before the pulse is finished, so the heating and cooling time constants are approximately equal. If this is undesired, raising the voltage used in the design procedure will give a smaller capacitance and a faster resonance frequency. The applied energy pulse will then be shorter and higher. For the current laboratory application losing half of the 7.1 mJ is not a problem so the design is left as it is.

The heat will quickly spread into the gas layer and the SiO_2 (silica) layer supporting the catalyst. Figure 6 shows how the heat spreads through the silica supporting layer in time. The conduction within the metal layer that lies beneath the silica layer is so fast that, even though the Si layer is 2 mm thick, it does not allow for high internal temperature gradients. The whole Si layer remains at approximately 300 K and has a maximum at the boundary with the silica layer of only 301.5 K. The whole temperature gradient between the Si layer and the catalyst layer exists in the supporting silica layer, which very quickly builds up a linear temperature profile, due to the fact that the silica has a much lower thermal conductivity than the catalyst or Si layer.

The other side of the catalyst layer loses heat to the gas layer. Figure 7 shows the heat profile within the gas layer. The heat conductivity for gas is even lower than the one for the top silica layer and thus much lower than that of the metallic catalyst layer. Just like with the supporting silica layer at the bottom, the majority of the temperature gradient at the top side of the catalyst will be in the layer

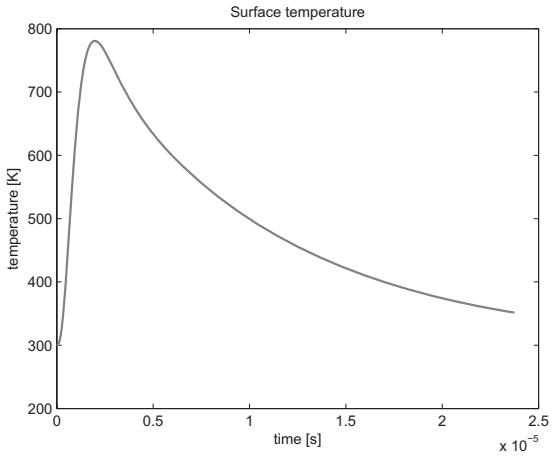


Fig. 5. Surface temperature when pulse is applied. The pulse causes the temperature to rise by about 500 K. The cooling time constant and the heating time constant are approximately equal for the parameters of table 1.

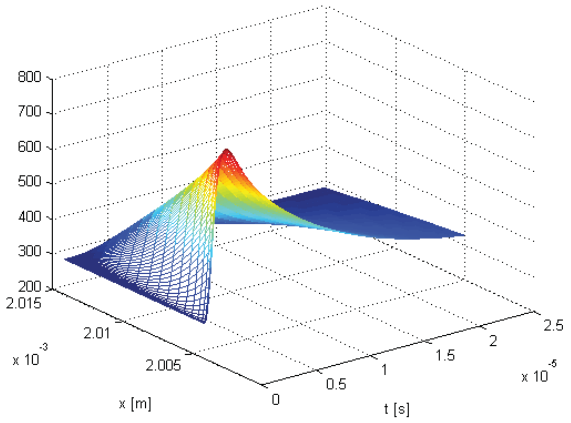


Fig. 7. The gas layer temperature profile over time. Like with the supporting SiO_2 layer a gradient leads to a linear profile, but the time taken for the linear profile to build up is longer.

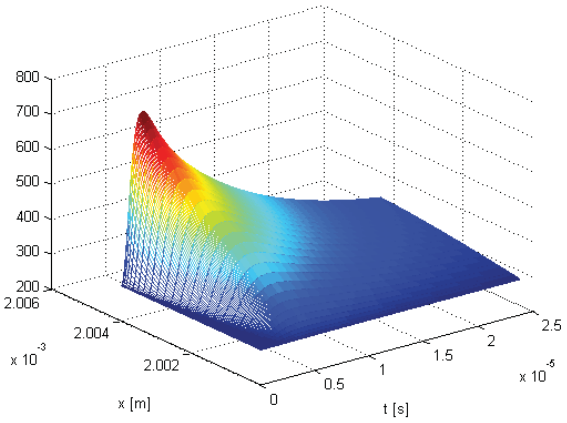


Fig. 6. Heat profile of the SiO_2 layer supporting the catalyst. A large temperature gradient over this layer quickly leads to a linear temperature profile.

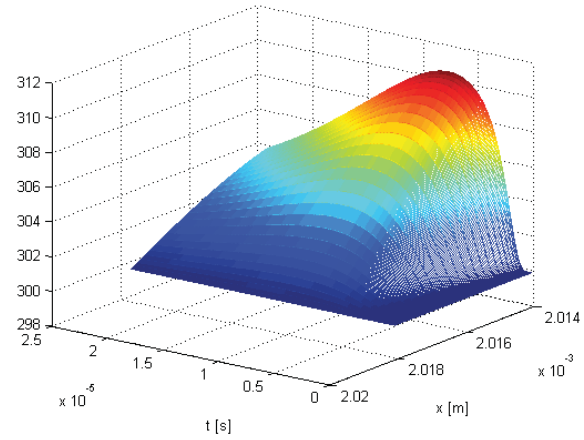


Fig. 8. Temperature profile within the top SiO_2 layer over time. The peak temperature is only about 10 K above the ambient temperature. Heat entering this layer from the gas layer is quickly transported to the top Si layer.

with the poorest heat conductivity which is the gas layer. Since the gas layer is thicker than the silica layer and even poorer in conducting heat, it takes a bit longer for the linear profile to build up. Due to the low conductivity of the gas, much less heat is lost to the gas than is lost through the supporting silica layer.

Figure 8 shows the top silica layer, which is heated through the gas layer. The maximum rise of temperature in the top silica layer is only 10 K even though the catalyst surface was heated by almost 500 K. The heat that ends up in the top silica layer is quickly lost to the top Si layer.

Just to confirm the results, the heat distribution within the platinum catalyst layer is shown in figure 9. From this figure it is clear that there exists almost no gradient within the catalyst layer, due to the superior heat conductivity of platinum.

The heating of the platinum layer is achieved through ohmic heating a resonant circuit with a high natural frequency and a low quality factor. By using higher voltages and adjusting the capacitance and inductance accordingly the heating can be made as fast as parasitic effects allow for. The cooling of the catalyst layer however cannot be forced, and the cooling time constant is created as an interplay between the different layers thicknesses and their values for thermal conductivity. By manipulating the thickness of the catalyst layer and the supporting silica layer the time constant for cooling can be shaped. A thickness of 100 nm for the catalyst layer and 4 µm for the supporting silica layer give a feasible time constant in this design.

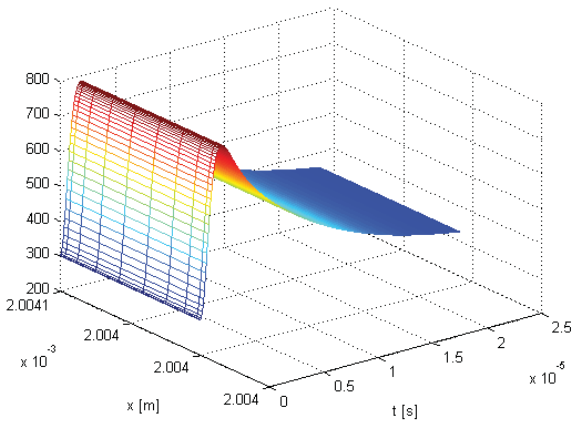


Fig. 9. The temperature profile within the catalytic layer. There exists almost no gradient within the catalyst layer, resulting in a flat profile.

4. POTENTIAL PITFALLS

Apart from the design variables already considered there are some miscellaneous design considerations that need to be included. This section mentions the most important ones.

4.1 Mechanical Strength

The heated metal layer will go up and down in temperature by hundreds of Kelvin. Metals normally expand when they become hot, and shrink when they cool down again. Unfortunately it is difficult to say whether this will physically break the layer or not. Since nobody ever tried this before there is no data on such high gradients in such thin supported metal layers. If mechanical strength is found to be a problem the pulses will need to be less intense.

4.2 Discharge Through SiO_2 Support

In the current design there is a 4 µm SiO_2 layer between the catalytic layer and the silicon support. The silicon support is electrically grounded for safety reasons. The catalytic layer is subjected to a high voltage. This voltage should not become so high that discharge occurs straight through the quartz layer because this will not only lead to pulse energy loss, but also break the device.

4.3 Electro Migration

When extremely high current densities are applied to any material, a phenomenon called electro migration will occur. The atoms of the metal start to physically move to one of the contacts, which in time will break the catalyst layer. To prevent this effect the current direction should be reversed between pulses.

4.4 Electromagnetic Interference

The high frequency currents are associated with a high frequency electromagnetic field. For any industrial application this field should be contained to such values that other electronic devices are not disturbed, and the radio spectrum is not polluted.

5. CONCLUSION

A new prototype reactor is presented to create extremely fast temperature pulses in heterogeneous catalysis. The authors intend to use this prototype to improve understanding of the effects of temperature pulsing on complex reaction schemes. The availability of such a reactor can help understanding the results of microwave enhanced catalysis as well. There appear to be no fundamental problems preventing a temperature pulse of 1000 K in less than 10^{-5} s, which is orders of magnitudes faster than what is known from literature.

REFERENCES

Bailey, J. (1973). Periodic operation of chemical reactors: A review. *Chemical Engineering Communications*, 1, 111–124.

Horn, F. and Lin, R. (1967). Periodic processes: A variational approach. *Industrial & Engineering Chemistry*, 6, 21–30.

Matsubara, M., Nishimura, Y., and Takahashi, N. (1973). Periodic operation of CSTR - I idealized control. *Chemical Engineering Science*, 28, 1369–1377.

Silveston, P. and Hudgins, R. (2004). Periodic temperature forcing of catalytic reactions. *Chemical Engineering Science*, 59, 4043–4053.

Silveston, P., Hudgins, R., and Renken, A. (1995). Periodic operation of catalytic reactors–introduction and overview. *Catalysis Today*, 25, 91–112.

Stolte, J., Vissers, J., Backx, T., and Bosgra, O. (2008). Modeling local/periodic temperature variation in catalytic reactions. In *IEEE International Conference on Control Applications*.

Trefethen, L. (2000). *Spectral Methods in Matlab*. SIAM.

Wan, J. (1993). Microwaves and chemistry: the catalysis of an exciting marriage. *Research on Chemical Intermediates*, 19, 147–158.

Wan, J., Tse, M., Husby, H., and Depew, M. (1990). High-power pulsed microwave catalytic processes: decomposition of methane. *Journal of Microwave Power and Electromagnetic Energy*, 25, 32–38.

Weideman, J. and Reddy, S. (2000). A matlab differentiation matrix suite. *ACM Transactions on Mathematical Software*, 26(4), 465–519.

Will, H., Scholz, P., and Ondruschka, B. (2004). Microwave-assisted heterogeneous gas-phase catalysis. *Chemical Engineering & Technology*, 27, 113.

Decision Oriented Bayesian Design of Experiments

Farminder S. Anand*, Jay H. Lee,**
Matthew J. Realff***

**School of Chemical & Biomolecular Engineering*
Georgia Institute of Technology, Atlanta, GA 30332 USA
(Tel:404-642-9151 ; e-mail: farminder.anand@ chbe.gatech.edu).
***(e-mail: jay.lee@chbe.gatech.edu)*
**** (e-mail: matthew.realff@chbe.gatech.edu)}*

Abstract: Experimental design is a fundamental problem in science and engineering. Traditional 'Design of Experiment' (DOE) approaches focus on minimization of variance. In this work, we propose a new "decision-oriented" DOE approach, which takes into account how the generated data, and subsequently the developed model, will be used in decision making. By doing so, the variance will be distributed in a manner such that its impact on the targeted decision making will be minimal. Our results show that the new decision-oriented experiment design approach significantly outperforms the standard D-optimal design approach. The new design method should be a valuable tool when experiments are conducted for the purpose of making R&D decisions.

Keywords: Decision making, optimal experiment design.

1. INTRODUCTION

Design of experiments (DOE) as a field has evolved over the period of last few decades. Its importance has grown significantly because of the increasing need to reduce the resource requirement for achieving the target. The targets historically perceived by the scientists in performing experiments have been driven towards understanding the underlying phenomenon or estimating the parameters. Consequently, the traditional DOE tools have been geared towards maximization of some measure of information or towards the minimization of the variance in the parameter estimates.

It is our opinion that this way of thinking over a long period of time has led the field to lose sight on the ultimate purpose of experiments in many applications. If one looks back into the history of the evolution of design of experiments one finds the answers in Bernardo (1979): "*Scientists typically does not have, nor can be normally expected to have, a clear idea of the utility of his results. An alternative is to design an experiment to maximize the expected information to be gained from it*". Bernardo (1979), further goes on proving that any *f* (function of the parameters, θ), in informational theoretical terms is '*garbling of θ*'. Hence follows the conclusion that maximization of information of θ is better than maximizing information on *f*.

This practice, while seeming logical, does not directly address the intended purpose of the experiments in many engineering applications. Today much of industrial research is driven by investment decisions, i.e., experiments are conducted with a specific objective in mind. For example, experiments can be conducted to aid decisions for the maximization of revenue function when investigating a new process or for the selection of a few processes among the large alternatives. In such scenarios, following the traditional route for design of experiments may be significantly suboptimal.

2. BACKGROUND

Traditionally there have been two major classes for the design of experiments (DOE) approaches: Classical approach and Bayesian approach. Historically, Classical DOE approaches like the factorial design have been more popular due to the computational complexities of the Bayesian approach. But recent developments in sampling techniques such as Markov Chain Monte Carlo (MCMC) (Kass (1998), Cowles (1996)) have rejuvenated the interest in the Bayesian approaches. In addition, the Bayesian approaches provide an added advantage of enabling the designer to incorporate the prior expert opinions. Hence, we will focus on the Bayesian approaches for design of experiments from here on.

To elaborate on the traditional Bayesian design strategies, we would follow Chaloner's (1995) approach, as it does justice to the inherent decision aspect hidden in the Bayesian approach. The idea of Bayesian DOE has evolved from information acquisition concepts in decision theory. Raiffa (1961) presented a decision theoretic approach for optimal information acquisition strategy using Expected Value of Information (EVOI) approach for investment decision problems. EVOI is defined as the expected difference between the expected posterior and prior utility, if one is to acquire information. Lindley (1956) introduced his seminal work on the use of Shannon information as a measure of information provided by an experiment. Following this, several authors (Stone (1959), DeGroot (1962) and Bernardo (1979)) presented a decision theoretic approach to experimental design, which was basically the maximization of EVOI with the utility function being replaced by Shannon information.

Consider a utility function (U), optimal decision (u) under posterior distribution, design matrix (η), parameters (θ) and observations (Y). Application of Lindley's EVOI maximization approach results in the maximization of expected pre-posterior utility as the expected value of the prior utility function is constant. The optimal expected pre-posterior utility is given in (1). Fig.1 demonstates how (1) can be solved numerically. Based on a given design (η) and the prior distribution of the parameters (θ_{prior}), potential observations (Y) are found via Monte Carlo simulation. For each of these potential realizations, posterior estimates (and covariances) for the parameters are obtained ($\theta_{Posterior}$) and corresponding to these posterior parameter estimates optimal decision variable (u) is estimated. Next step is to calculate the posterior expected utility(U) value corresponding to each of these potential realizations (Y). The average of the posterior utility values for each potential realization (Y) gives a utility of the design (η). The design that maximises this average utlity value is the optimal design.

$$U(\eta^*) = \max_{\eta} \int_{y} \max_{u} \int_{\Theta} U(\theta,\eta,Y,u)\, p(\theta \mid Y,\eta)\, p(Y \mid \eta)\, d\theta\, dY \quad (1)$$

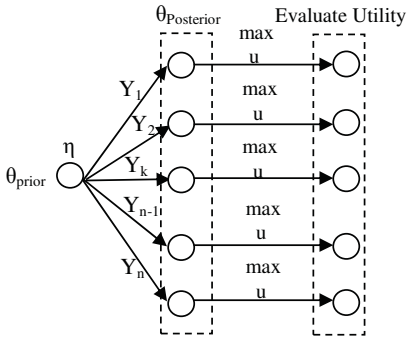


Fig. 1. Demonstrating the calculation of optimal design based on Lindley's EVOI concept.

Now, if one considers the Shannon information as the utility function, as suggested by Lindley, the above simplifies to (2). As both Fig. 2 and (2) show the calculations become much more tangible as the 'max' step drops out.

$$U(\eta^*) = \max_{\eta} \int_{y} \int_{\Theta} \log\{p(\theta \mid Y,\eta)\}\, p(Y,\theta \mid \eta)\, d\theta\, dY \quad (2)$$

The rest of the Bayesian DOE methodologies, follow a similar line as (2) with some small changes to the utility function. In a broad sense, there exist three categories of Bayesian DOE approaches. First is the information maximization approach, which consists of maximizing the Kullback - Leibler distance between the prior and the posterior distribution. This approach consists of D-optimal and Ds-optimal designs. The second category is the set of designs, where the objective is to obtain a point estimate of the parameter values. This category consists of A-optimal and C-optimal designs. The third category is the minimax type of designs, where the objective is to minimize the maximum

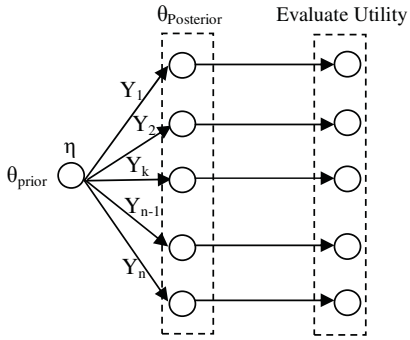


Fig. 2. Demonstrating the calculation of optimal design based on Shannon information criterion.

possible variance for all the linear combinations of the parameters under consideration. These various designs are further explained in details as follows:

D - Optimal: Maximize information gain for the parameters (Uses Kullback-Leibler distance between the prior and posterior distribution as a measure of gain in information).

Ds-optimality: Maximize gain in Shannon information of Ψ (= $S^T\theta$), where 'S' is a known constant vector.

A- Optimal: The objective of the experiment is to obtain a point estimate of the parameters. A design is chosen to maximize the following utility function:

$$U(\eta) = -\int\int (\theta - \hat{\theta})^T A (\theta - \hat{\theta}) p(y,\theta \mid \eta)\, d\theta\, dy \quad (3)$$

Here 'A' is a symmetric nonnegative definite matrix. This design minimizes expected squared error of loss for estimating $C^T\theta$ or Minimizing square error of predicting at C, where C is not necessarily a fixed and a distribution is specified on it.

C- Optimal: Special case of A-optimality, where C is a constant.

E- Optimal: It is a minimax approach for variance. The maximum posterior variance of all possible normalized linear combinations of parameter estimates is minimized. An, E-optimal design minimizes:

$$\sup_{\|c\|=\omega} c^T \left(\eta^T\eta + R\right)^{-1} c = \omega^2 \lambda_{\max}\left[\left(\eta^T\eta + R\right)^{-1}\right] \quad (4)$$

G- Optimal: Closely related to E-optimal deign is G-optimal design, which minimizes $\sup_{x \in D} x^T \left(\eta^T\eta + R\right)^{-1} x$. An equivalence theorem [see Atkinson (1992)] states that continuous G-optimal designs are numerically identical to a corresponding continuous D-optimal design.

It is important to note that, among the above mentioned designs, D-, Ds- A- and C-optimal design have a utility function, which justifies its decision theoretic sense. On the other hand, E- and G-optimal designs though are considered

Bayesian design don't have any decision-theoretic sense, Chaloner (1995).

The rest of the document is structured as follows: Section 3 discusses in more detail the setup for the decision oriented design, section 4 presents the numerical results, and section 5 concludes the paper.

3. SELECTION/REJECTION DESIGN

As elaborated in the previous section the traditional design criterions either try to maximize the information gain or minimize the variance. Consider the case when the objective of the experimentation is to select/reject processes from a large set of potential processes. In this scenario traditional overall variance reduction design techniques may not be the optimal solution. For example, assume that the selection criterion is based on a cut-off value of operating profit margins, say \$10M/yr and processes that have operating profit margins equal or above the cut-off are worth pursuing. The question at this juncture may be: Should one be more focused towards reducing the overall parameter uncertainty or towards designing experiment strategies that directly target this objective?

In order to design experiments focused on this target, we propose to design experiments that maximize the expected operating profit margin. The premise here is that the designs that try to obtain the maximum operating profit margins would inherently be able to obtain values closer to the true optimal operating profit margin values. In order to obtain such a DOE we substitute the operating profit margin function in place of the utility function 'U' in (1).

3.1 Problem Formulation

Assume an initial model structure and prior estimates for the process models are available from the prior experimental results. The decision-maker wants to perform more experiments to select the few processes with the most potential.

Assume the yield (Y_1) of the process has a linear model, $Y_1 = \tilde{X}^T\tilde{\theta}_1 + \varepsilon$, where $\tilde{X}$ is the vector of the operating conditions to be optimized and ε is the Gaussian noise, N(0,σ) with known variance(σ^2). We assume that the quality of the product also varies linearly, $Y_2 = \tilde{X}^T\tilde{\theta}_2 + \varepsilon$, with the operating conditions and the target quality is μ. We consider the operating profit margin function (*f*) in Eq. (5), which depends linearly on the yield value, has a quadratic penalty for the quality deviation, and a quadratic penalty (Q) for higher operating conditions.

$$f = \alpha * Y_1 - \beta * (Y_2 - \mu)^2 - \frac{1}{2}\tilde{X}^T * Q * \tilde{X} \qquad (5)$$

To obtain a DOE which maximizes the operating profit margins, we substitute *f*, operating profit margin function in place of the utility 'U' in (1) and follow the algorithm as explained in Fig.1 and section 2.

To evaluate the new design criterion, we consider $\tilde{X} = [x_1, x_2]^T$ to be a two dimensional vector and hence both the prior parameter estimates $\tilde{\theta}_1 = [\theta_{1,1}, \theta_{1,2}]^T$ & $\tilde{\theta}_2 = [\theta_{2,1}, \theta_{2,2}]^T$ are also two dimensional vectors. We consider the range of the operating conditions to be in the range of [1e-5, 10]. We consider the prior estimates of the parameters (θ_1 and θ_2) to be normal distributions with mean $\bar{\theta}_1 = [\bar{\theta}_{1,1}, \bar{\theta}_{1,2}]^T$, $\bar{\theta}_2 = [\bar{\theta}_{2,1}, \bar{\theta}_{2,2}]^T$ and covariance matrices $\Sigma_{\theta 1}$ and $\Sigma_{\theta 2}$ respectively.

In order to statistically evaluate the performance of our DOE approach against the traditional D-optimal DOE approach, we consider the following distributions for the parameter values:

$$\bar{\theta}_{1,1} \sim U[-100, 100] \qquad (6)$$

$$\bar{\theta}_{1,2} \sim U[\max(-100, -\bar{\theta}_{1,1}), 100] \qquad (7)$$

$$\bar{\theta}_{2,1} \sim U[-100, 100] \qquad (8)$$

$$\bar{\theta}_{2,2} \sim U[\max(-100, -\bar{\theta}_{2,1}), 100] \qquad (9)$$

$$\Sigma_{\theta 1} = [(0.1*\bar{\theta}_{1,1})^2 \;\; 0 \;;\; 0 \;\; (0.1*\bar{\theta}_{1,2})^2] \qquad (10)$$

$$\Sigma_{\theta 2} = [(0.1*\bar{\theta}_{2,1})^2 \;\; 0 \;;\; 0 \;\; (0.1*\bar{\theta}_{2,2})^2] \qquad (11)$$

$$\sigma = \sqrt{\min((0.1*\bar{\theta}_{1,1})^2, (0.1*\bar{\theta}_{1,2})^2)} \qquad (12)$$

$$\mu = U[0.5*1e\text{-}5*\min(\bar{\theta}_{2,1}, \bar{\theta}_{2,2}),\ 1.5*10*\max(\bar{\theta}_{2,1}, \bar{\theta}_{2,2})] \qquad (13)$$

The idea behind choosing the above parameter space is not only to have a sufficiently broad range of the parameter space but also to have some realistically sensible parameter values. The quadratic penalty matrix (14), Q, for higher operating conditions is chosen appropriately so that it is both positive definite and a practically reasonable value.

$$Q = [q_{11}\ q_{12};\ q_{21}\ q_{22}], \text{ where} \qquad (14)$$

$$q_{11} = U[1e^{-5}, \alpha * |(\bar{\theta}_{1,1} + \bar{\theta}_{1,2})/2|] \qquad (15)$$

$$q_{22} = U[1e^{-5}, \alpha * |(\bar{\theta}_{2,1} + \bar{\theta}_{2,2})/2|] \qquad (16)$$

$$q_{12} = U[1e^{-5}, \sqrt{(q_{11}*q_{22})}] \qquad (17)$$

$$q_{21} = U[1e^{-5}, \; q_{11}*q_{22}/q_{12}] \qquad (18)$$

Lastly the true parameter values, unknown to the decision-maker, are considered to be drawn randomly from the prior parameter distributions.

3.2 Solution Approach

To obtain the optimal design solution for the above mentioned problem, we need to solve (19). In (19) Y is the two dimensional vector $[Y_1, Y_2]^T$, each term corresponding to the yield and the quality value. And Θ is the vector of the corresponding parameters for the yield ($\tilde{\theta}_1 = \left[\theta_{1,1}, \theta_{1,2}\right]^T$) and quality ($\tilde{\theta}_2 = \left[\theta_{2,1}, \theta_{2,2}\right]^T$) respectively. Algorithm to calculate the optimal design via (19) is shown in Fig. (3).

$$U(\eta^*) = \max_{\eta} \int_y \max_u \int_\Theta f(\theta, \eta, Y, u)\, p(\theta \mid Y, \eta)\, p(Y \mid \eta)\, d\theta dY \quad (19)$$

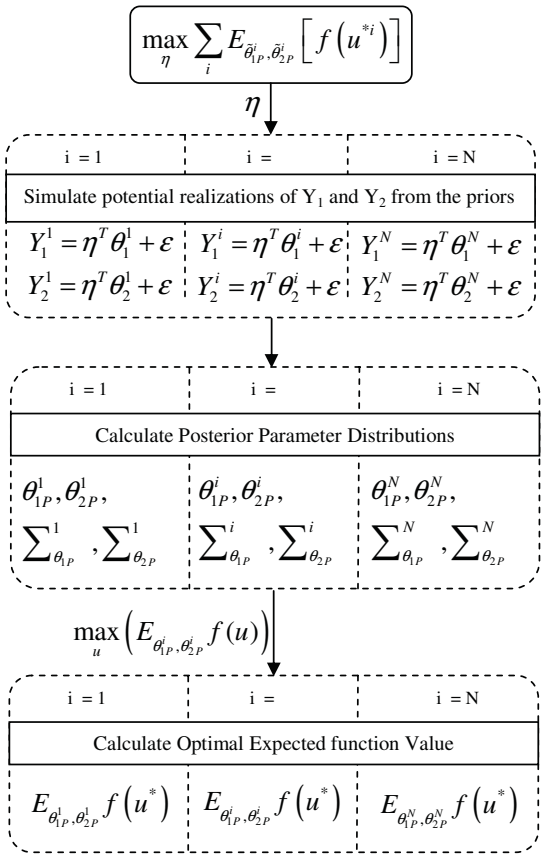


Fig. 3. Algorithm to calculate the optimal decision oriented design of experiment.

The calculation algorithm consists of two stages of optimization. The outer optimization is for selecting the optimal design and the inner optimization is for obtaining the optimal posterior operating conditions. The details for evaluating a given design 'η' are explained as follows:

Step 0: Assume an initial design 'η'
Step 1: Based on the given design and the prior distributions for the parameters $\tilde{\theta}_1 = \left[\theta_{1,1}, \theta_{1,2}\right]^T$ and $\tilde{\theta}_2 = \left[\theta_{2,1}, \theta_{2,2}\right]^T$ generate potential realizations of Y_1^i and Y_2^i for i = 1, 2,..., N (we consider N = 500).

Step 2: For each given realization estimate the posterior mean and covariance matrix for the parameters $\tilde{\theta}_1$ and $\tilde{\theta}_2$.

Step 3: For each posterior distribution estimate, obtain the optimal operating condition and hence the optimal function '*f*' value. Since the function '*f*' has nice structure in our case, the optimization has an analytical solution. But due to the constraints on the operating conditions, the optimal operating conditions are either at the boundary of the constraints or are given by the analytical solution.

Step 4: Calculate the expected value of the optimal function '*f*' value for each of the distributions.

Step 5: Calculate the average of all the optimal expected function values calculated in Step 4.

The value obtained in Step 5 is the value signifies potential of the given design ''. In order to obtain the optimal design, maximization is performed over the design space. This maximization is performed using the inbuilt function '*fmincon*' in MATLAB.

4. RESULTS

To compare the results given by our new design approach and the D-optimal design approach, we took 100,000 runs for different randomly sampled parameter values. To measure the performance of different designs, we measure the closeness of the predicted operating profit margin value to the true optimal operating profit margin value. The percentage of times the true value is closer to the predicted value by a design is reported as the 'Performance Index' of that design.

Table 1 Comparison of performance of new- and D-optimal designs for the 10% noise case.

Type of Prior Distribution	'Performance Index'	
	New Design	D Design
Strongly Informative	45.2290	25.9920
Informative	53.5120	22.8120
Mildly Informative	58.2250	21.1130
Un-Informative	63.5690	17.3060

In order to check if the kind of prior distribution has an effect on the performance of the new-design approach, we measure four levels of prior distributions. An '*Informative*' prior distribution is the one with square root of the diagonal elements of the prior covariance matrix being 10% of the prior mean of the respective parameter. This kind of prior distribution is the one we have shown in (10) and (11). A '*Strongly Informative*' prior distribution is the one with a small covariance and we depict it by replacing the 0.1 values

by 0.05 in (10) and (11). A '*Mildly Informative*' prior distribution is the one with a relatively high variance and is depicted by replacing 0.1 values by 0.15 in (10) and (11). A '*Un-Informative*' prior distribution is the one with a relatively high variance and is depicted by replacing 0.1 values by 0.30 in (10) and (11). The comparison of the performance measure for our decision-oriented and the D-optimal design is shown in Table 1. The above results clearly show ~20-35% improvement in the prediction power of the Decision-oriented design compared to the D-optimal design of experiments.

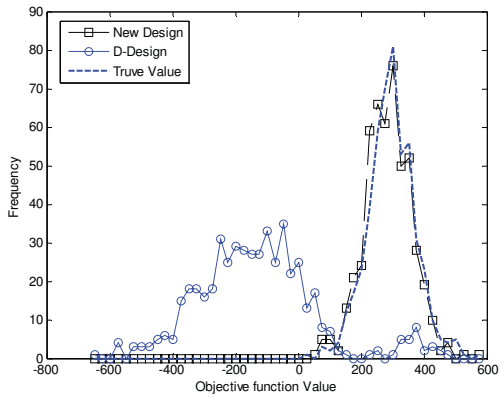


Fig. 4. Histogram comparing the prediction of the Decision oriented and D-optimal design to the true objective value for a '*Mildly Informative*' prior distribution.

Table 2 Comparison of performance of the Decision-oriented design and the D-optimal designs for the 5% noise case

Type of Prior Distribution	'Performance Index'	
	New Design	D Design
Strongly Informative	52.8330	23.3690
Informative	60.5130	18.2910
Mildly Informative	63.9620	16.4300
Un-Informative	67.3060	14.0870

To give more insight to the results we plot the histogram of the optimal operating margin values for the decision oriented and D-optimal design along with the optimal operating margin values (determined assuming that the 'true' parameter values are known), for a particular set of parameter values with 500 different 'true' parameter values being sampled from the prior distribution. To be precise, these are the operating margins for the 'true' plant (with 'true' parameter values) with the optimal operating conditions determined based on the parameter estimates resulting from the respective DOEs. Fig 4 shows the histogram for a '*Mildly Informative*' prior distribution, with 500 'true' parameter values sampled from the prior distribution.

Similarly Fig. 5, Fig. 6 and Fig. 7 show the histogram plots for the '*Informative'*, '*Strongly Informative*' and '*Un-Informative*' prior distributions respectively. Fig 4, 5, 6 & 7 clearly demonstrate the better performance of the decision-oriented DOE strategy compared to traditional D-optimal design strategy.

Table 3 Comparison of performance of the Decision-oriented design and the D-optimal designs for the 15% noise case

Type of Prior Distribution	'Performance Index'	
	New Design	D Design
Strongly Informative	39.5190	25.4060
Informative	49.3160	24.8090
Mildly Informative	53.5440	22.6560
Un-Informative	60.2590	19.1200

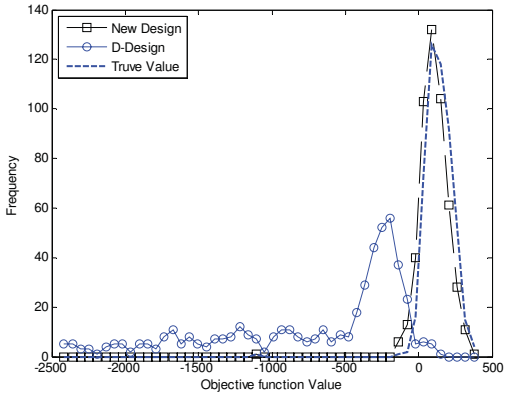


Fig. 5. Histogram comparing the prediction of the Decision oriented and D-optimal design to the true objective value for an '*Informative*' prior distribution.

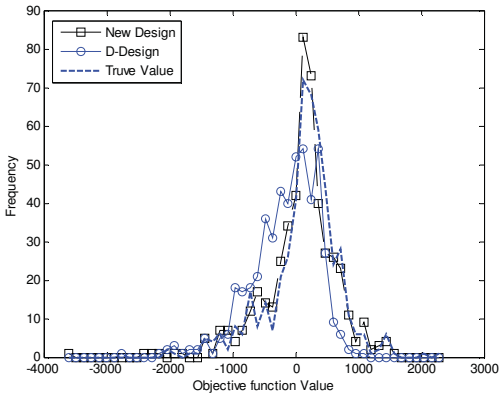


Fig. 6. Histogram comparing the prediction of the Decision oriented and D-optimal design to the true objective value for a '*Strongly Informative*' prior distribution.

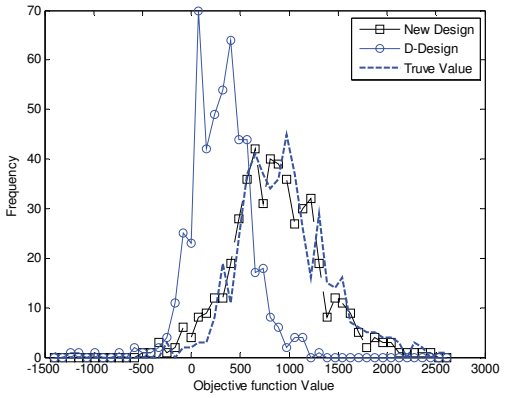


Fig. 7. Histogram comparing the prediction of the Decision oriented and D-optimal design to the true objective value for a '*Un-Informative*' prior distribution.

In order to check if noise has any significant impact on the performance of the decision-oriented designs, we vary the noise measured by the variance of the Gaussian noise in (12). In comparison to the initial noise of 10% as depicted by the value '0.1' in (12) we test two other levels of noise 5% and 15% , which correspond to changing '0.1 value in (12) to '0.05' and '0.15' respectively. The results for the 5% and the 15% noise cases are shown in the Table 2 and Table 3 respectively. The results clearly demonstrate that the decision-optimal design outperforms the D-optimal design regardless of the noise level.

5. CONCLUSIONS

We have introduced a new decision oriented design of experiment strategy, which significantly improves the prediction of a process's optimal objective function value compared to that of a D-optimal design of experiment strategy. These types of DOE strategies are expected to be of significant importance in improving the R&D decisions, especially in bio-fuel related research where one faces multiple process alternatives. Moreover, in addition to the design criterion considered in this work, one can consider alternative Acceptance/Rejection design criterion. For example, in the problem discussed in this work, we were mainly concerned with the mean value of '*f*', but an alternat-

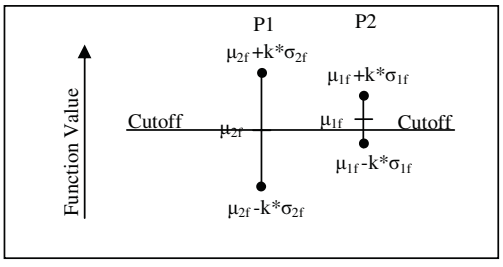


Fig. 8. An alternative Acceptance/Rejection criterion

-ive design criterion can be based on both the mean and the variance of '*f*' along with a cut-off value. Consider two processes P1 and P2, shown in Fig. 8. The selection criterion of the decision maker for these processes is that $\mu - k*\sigma$ be greater than the 'Cutoff' and the rejection criterion being that $\mu + k*\sigma$ be less than the 'Cutoff', where μ is the posterior mean and σ is the posterior standard deviation of the objective function '*f*'. To design experiments for selection/rejection of processes based on this type of criterion can be done by maximising $|\Sigma\delta_i|$, where δ_i is defined as follows:

$$\delta_i = \begin{cases} +1, if\ \mu_{fi} - k*\sigma_i > Cutoff \\ -1, if\ \mu_{fi} + k*\sigma_i < Cutoff \\ 0, Otherwise \end{cases}$$

The subscript 'i' represents the potential random samples with value ranging form i = 1, 2.., N, as explained earlier in section 3.2. Similarly various other design criterions can be created based on the decision maker's objective function. We will evaluate these and similar acceptance/rejection decision criterions in our future work.

REFERENCES

Atkinson, A.C. and Donev, A.N. (1992). *Optimum Experimental Designs*, Pg. 114, Ch.10. Oxford Science Publications, United States

Bernardo, J.M.(1979). Expected information as expected utility. *Annals of Statistics*, 7 686-690

Chaloner, K. and Verdinelli, I. (1995). Bayesian Experimental Design: A Review, *Statistical Science*, 10 273-304

Cowles, Mary K. and Carlin, Bradley P. (1996), Markov Chain Monte Carlo Convergence Diagnostics: A comparative Review, *Journal of the American Statistical Association*, 91 883-904.

DeGroot, M.H. (1962). Uncertainty, information and sequential experiments, *The Annals of Mathematical Statistics*, 33 404-419

Kass, Robert E., Carlin, Bradley P., Gelman, Andrew, and Neal, Radford M., Markov Chain Monte Carlo in Practice: A Roundtable Discussion, *The American Statistician*, 52 93-100.

Lindley, D.V. (1956). On the measure of information and provided by an experiment. *Annals of Statistics*,27 986-1005

Raiffa, H. and Schlaifer, R. (1961). Applied statistical decision theory, Division of Research, Graduate School of Business Administration, Harvard University, Boston, United States

Stone, M. (1959). Application of a measure of information to the design and comparison of regression experiment. *The Annals of Mathematical Statistics, 30 55-70*

Correlation-Based Pattern Recognition and Its Application to Adaptive Soft-Sensor Design

Koichi Fujiwara * **Manabu Kano** * **Shinji Hasebe** *

* *Dept. of Chemical Engineering, Kyoto University, Katsura Campus, Nishikyo-ku, Kyoto 615-8510, Japan (e-mail: manabu@cheme.kyoto-u.ac.jp)*

Abstract: Although soft-sensors have been widely used for estimating product quality or other key variables, they do not always function well in practice due to changes in process characteristics. The Correlation-based Just-In-Time (CoJIT) modeling has been proposed to cope with changes in process characteristics. In the CoJIT modeling, the samples used for local modeling are selected on the basis of correlation together with distance, since changes in process characteristics are expressed as the difference of the correlation. In addition, the individuality of production devices should be considered when they are operated in parallel. However, the CoJIT modeling cannot cope with the individuality of production devices because it is only applicable to time-series data. In the present work, a new pattern recognition method, referred to as the Nearest Correlation (NC) method is proposed, and it selects samples whose correlations are similar to the query. In addition, the proposed NC method is integrated with the CoJIT modeling. The advantages of the proposed CoJIT modeling with the NC method are demonstrated through a case study of a parallelized CSTR process.

Keywords: Soft-sensor, Estimation, Prediction, Just-In-Time modeling, Pattern recognition, Principal component analysis

1. INTRODUCTION

A soft-sensor, or a virtual sensor, is a key technology for estimating product quality or other important variables when on-line analyzers are not available. Partial least squares (PLS) regression and artificial neural network (ANN) have been widely accepted as useful techniques for soft-sensor design (Kano and Nakagawa (2008), Mejdell and Skogestad (1991), Kresta et al. (1994), Kano et al. (2000), Kamohara et al. (2004) and Radhakrishnan and Mohamed (2000)). In addition, the application of subspace identification (SSID) to soft-sensor design has been reported in Amirthalingam and Lee (1999) and Kano et al. (2008) for achieving higher estimation performance.

Generally, building a high performance soft-sensor is very laborious, since input variables and samples for model construction have to be selected carefully and parameters have to be tuned appropriately. In addition, even if a good soft-sensor is developed successfully, its estimation performance deteriorates as process characteristics change. In chemical processes, for example, process characteristics are changed by catalyst deactivation or fouling. Such a situation may deteriorate product quality. Therefore, maintenance of soft-sensors is very important in practice to keep their estimation performance. Ogawa and Kano (2008) indicate that soft-sensors should be updated as the process characteristics change, and also manual, repeated construction of them should be avoided due to its heavy workload.

To update statistical models automatically when process characteristics change, recursive methods such as recursive PLS (Qin (1998)) were developed. These methods can adapt models to new operating conditions recursively. However, when a process is operated within a narrow range for a certain period of time, the model will adapt excessively and will not function in a sufficiently wide range of operating conditions. In addition, recursive methods cannot cope with abrupt changes in process characteristics.

On the other hand, the individuality of production devices should be taken into account. In semiconductor processes, for example, parallelized production devices are used, and they have different characteristics even if their catalog specifications are the same. Therefore, a soft-sensor developed for one device is not always applicable to another device, and it is very laborious to customize soft-sensors according to their individuality.

The Just-In-Time (JIT) modeling has been proposed to cope with process nonlinearity (Bontempi et al. (1999) and Atkeson et al. (1997)) and changes in process characteristics (Cheng and Chiu (2004)). In the JIT modeling, a local model is built from past data around the query only when an estimate is required. The JIT modeling is useful when global modeling does not function well. However, its estimation performance is not always high because the samples used for local modeling are selected on the basis of the distance from the query and the correlation among variables is not taken into account. How should we determine the samples used for local modeling to build

a highly accurate statistical model? Distance is not the most important. A good model cannot be developed when correlation among input-output variables is weak, even if the distance between samples is very small. Conversely, a very accurate model can be developed when the correlation is strong even if the distance is large.

Recently, a new JIT modeling method based on the correlation among variables, referred to as the correlation-based JIT (CoJIT) modeling, has been proposed by Fujiwara et al. (2008). In the CoJIT modeling, the samples used for local modeling are selected on the basis of correlation together with distance. The CoJIT modeling can cope with abrupt changes of process characteristics and also achieve high estimation performance. However, it is applicable to only time-series data because it uses moving windows to generate data sets for local modeling. In other words, the original CoJIT modeling cannot generate a data set consisting of such data that represent characteristics of a query sample and are obtained from various devices operated in parallel.

To make the CoJIT modeling applicable to soft-sensor design for parallelized production devices, samples obtained from various devices have to be discriminated on the basis of the correlation among variables. This discrimination problem is one of the unsupervised pattern recognition problems because the teacher signal is not used for sample classification.

The Nearest Neighbor (NN) method and k-means method are well-known conventional unsupervised pattern recognition algorithms. The NN method can detect samples that are similar to the query, and k-means method can cluster samples without the teacher signal. However, they are distance-based methods and do not take into account the correlation among variables. Self organizing map (SOM) also has been used as an unsupervised pattern recognition method (Kohonen (2001)). SOM is a machine learning process that imitates the brain ' s learning process, and it not only can classify samples but also can visualize high dimensional data. However, it requires high computational load, and the preprocessing data are complicated.

In the present work, to cope with the individuality of production devices as well as changes in process characteristics, a new unsupervised pattern recognition method based on the correlation among variables, referred to as the Nearest Correlation (NC) method, is proposed. The proposed NC method can detect samples that have correlation similar to the query on the basis of sample geometry. In addition, the proposed NC method is integrated with the CoJIT modeling. The usefulness of the integration is demonstrated through a case study of a parallelized CSTR process.

2. INDICES OF CORRELATION

In this section, several measures for quantifying correlation among variables are briefly explained.

2.1 Correlation coefficient

The correlation coefficient $C_{i,j}$ can be used as an index of the similarity between two vectors $\boldsymbol{x}_i$ and $\boldsymbol{x}_j \in \Re^M$.

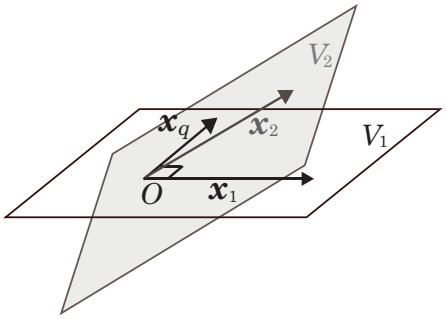


Fig. 1. An example of vector geometry in 3-dimensional space

$$C_{i,j} = \frac{\boldsymbol{x}_i^{\mathrm{T}} \boldsymbol{x}_j}{||\boldsymbol{x}_i|| ||\boldsymbol{x}_j||} = \cos\theta \tag{1}$$

where, θ is the angle between two vectors.

Suppose that the samples in the three-dimensional data consist of two classes K_1 and K_2, and samples belonging to classes K_1 and K_2 span the two-dimensional linear subspaces V_1 and V_2, respectively, as shown in Fig. 1.

Now, the query $\boldsymbol{x}_q$ is newly measured, and its class should be identified as K_1 or K_2. The correlation coefficients can be used as the index of sample discrimination. For example, $\boldsymbol{x}_1 \in K_1$ and $\boldsymbol{x}_2 \in K_2$ are selected from each class in a random manner, and the correlation coefficients between $\boldsymbol{x}_q$ and them are calculated respectively, and the class including the sample with the largest correlation coefficient can be identified as the class of $\boldsymbol{x}_q$.

In many cases, however, this method is inappropriate. In Fig. 1, the selected sample $\boldsymbol{x}_1$ and the query $\boldsymbol{x}_q$ are orthogonal to each other even though both vectors belong to K_1. In such a case, $\boldsymbol{x}_q$ is identified as an element of K_2 because the correlation coefficient between $\boldsymbol{x}_q$ and $\boldsymbol{x}_2$ is larger than the correlation coefficient between $\boldsymbol{x}_q$ and $\boldsymbol{x}_1$.

2.2 The Q statistic

In this work, the Q statistic is used as an index of sample discrimination.

The Q statistic is derived by principal component analysis (PCA), and it expresses the distance between the sample and the subspace spanned by principal components (Jackson and Mudholkar, 1979). The Q statistic is defined as

$$Q = \sum_{m=1}^{M} (x_m - \hat{x}_m)^2 \tag{2}$$

where x_m and $\hat{x}_m$ are the mth measurement and its estimate by the PCA model, respectively. The Q statistic is a measure of dissimilarity between the sample and the modeling data from the viewpoint of the correlation among variables.

In addition, to take into account the distance between the sample and the origin, Hotelling's T^2 statistic can be used. The T^2 statistic is defined as

$$T^2 = \sum_{r=1}^{R} \frac{t_r^2}{\sigma_{t_r}^2} \tag{3}$$

where σ_{t_r} denotes the standard deviation of the rth score t_r. The T^2 statistic expresses the normalized distance from the origin in the subspace spanned by principal components. The Q and T^2 statistics can be integrated into a single index for sample selection as proposed by Raich and Cinar (1994):

$$J = \lambda T^2 + (1-\lambda)Q \tag{4}$$

where $0 \leq \lambda \leq 1$.

3. NEAREST CORRELATION METHOD

The NN method and the k-means method can discriminate or cluster samples on the basis of the distance without a teacher signal. However, they do not take into account the correlation among variables. In this section, a new unsupervised pattern recognition method based on the correlation among variables, referred to as the nearest correlation (NC) method, is proposed. In the proposed NC method, sample geometry is used for sample discrimination.

3.1 Concept of the NC method

Suppose that the hyper-plane P in Fig. 2 (left) expresses the correlation among variables and the samples on P have the same correlation. Although samples $\boldsymbol{x}_1$ to $\boldsymbol{x}_5$ have the same correlation and they are on P, samples $\boldsymbol{x}_6$ and $\boldsymbol{x}_7$ have different correlation from the others. The NC method aims to detect samples whose correlation is similar to the newly measured query $\boldsymbol{x}_q$. In this example, $\boldsymbol{x}_1$ to $\boldsymbol{x}_5$ on P should be detected.

At first, the whole space is translated so that the query becomes the origin. That is, $\boldsymbol{x}_q$ is subtracted from all samples $\boldsymbol{x}_i (i = 1, 2, \cdots, 7)$. Since the hyper-plane P is translated to the plane containing the origin, it becomes the linear subspace V.

Next, a line connecting each sample and the origin is drawn. Suppose another sample can be found on this line. In this case, $\boldsymbol{x}_1$-$\boldsymbol{x}_4$ and $\boldsymbol{x}_2$-$\boldsymbol{x}_3$ satisfy such a relationship as shown in Fig. 2 (right). The correlation coefficients of these pairs of samples must be 1 or -1. On the other hand, $\boldsymbol{x}_6$ and $\boldsymbol{x}_7$ that are not the elements of V cannot make such pairs. Therefore, the samples of the pairs whose correlation coefficients are ± 1 are thought to have the same correlation as $\boldsymbol{x}_q$.

However, $\boldsymbol{x}_5$ that does not make a pair cannot be detected by this method even though it is on V. To detect $\boldsymbol{x}_5$, a linear subspace is derived from the selected pairs by using PCA, and the derived linear subspace corresponds to V.

Finally, the Q statistics for all samples $\boldsymbol{x}_i$ $(i = 1, 2, \cdots, 7)$ are calculated by using the PCA model expressing V. The samples with small Q statistics are located close to the linear subspace V, and such samples have correlation similar to the query. Although $\boldsymbol{x}_5$ cannot be detected in the previous step, it can be detected in this step because its Q statistic is 0. On the other hand, $\boldsymbol{x}_6$ and $\boldsymbol{x}_7$ are not detected in this step since they have large Q statistics.

In addition, the T^2 statistic can be used to take into account the distance from the origin. In the present work,

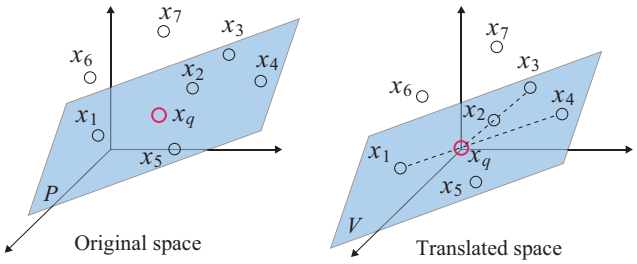


Fig. 2. An example of the procedure of the NC method

J in Eq. (4) is used as the index for sample selection. The samples with small J are selected as the samples similar to the query.

In the implementation of the above procedure, the threshold of the correlation coefficient γ $(1 \geq \gamma > 0)$ has to be used since there are no pairs whose correlation coefficient is strictly ± 1. That is, the pairs should be selected when the absolute values of their correlation coefficients are larger than γ.

3.2 Algorithm of the NC method

Assume that the samples stored in the database are $\boldsymbol{x}_n \in \Re^M$ $(n = 1, 2, \cdots, N)$ and the query is $\boldsymbol{x}_q \in P$ $(\dim(P) = R)$. The samples belonging to P should be detected in a manner similar to $\boldsymbol{x}_q$. The algorithm of the proposed NC method is as follows:

(1) Set R, $\gamma(1 \geq \gamma > 0)$, $\delta(\delta > 0)$ and K or $\bar{J}$.
(2) $\boldsymbol{x}'_n = \boldsymbol{x}_n - \boldsymbol{x}_q$ for $n = 1, 2, \cdots, N$.
(3) Calculate the correlation coefficients $C_{k,l}$ between all possible pairs of $\boldsymbol{x}'_k$ and $\boldsymbol{x}'_l$ $(k \neq l)$.
(4) Select the pairs satisfying $|C_{k,l}| \geq \gamma$, and set the number of the selected pairs S.
(5) If $S < R$, then $\gamma = \gamma - \delta$ $(\delta > 0)$ and return to step 4. If $S \geq R$, then go to the next step.
(6) Arrange the samples of the pairs selected in step 4 as the rows of the matrix $\boldsymbol{X}'$.
(7) Derive the linear subspace V from $\boldsymbol{X}'$ by using PCA. The number of principal components is R.
(8) Calculate the index J of $\boldsymbol{x}'_n$, and $J_n = J$ for $n = 1, 2, \cdots, N$.
(9) Detect the first K samples in ascending order of J_n or the samples whose J_n is smaller than $\bar{J}$ as samples similar to the query $\boldsymbol{x}_q$, where $\bar{J}$ is the threshold.

In step 5, when S is smaller than R, the threshold γ has to be relaxed to increase the number of selected pairs since the linear subspace V is not spanned by the samples of the selected pairs. R can be used as the tuning parameter.

3.3 Numerical example

The discrimination performance of the proposed NC method is compared with that of the NN method through a numerical example. In this example, data consist of three classes that have different correlations, and the samples belonging to the same class as the query should be detected. The discrimination rate is defined as

$$\text{Discrimination Rate [\%]} = \frac{L}{K} \times 100 \tag{5}$$

where K is the number of detected samples and L ($L \leq K$) is the number of samples that belong to the same class as the query among the detected samples. Samples in each of three classes are generated by using the following equation.

$$\boldsymbol{x}_i = \boldsymbol{A}_i \boldsymbol{s} + \boldsymbol{n} \ (i = 1, 2, 3) \tag{6}$$

$$\boldsymbol{s} = [s_1 \ s_2 \ s_3]^T \tag{7}$$

$$\boldsymbol{n} = [n_1 \ n_2 \ n_3]^T \tag{8}$$

where $\boldsymbol{A}_i$ is a coefficient matrix, $s_i \sim N(0, 10)$ and $n_i \sim N(0, 0.1)$. $N(m, \sigma)$ is the random number following the normal distribution whose mean is m and standard deviation is σ. The coefficient matrices are as follows:

$$\boldsymbol{A}_1 = \begin{bmatrix} 1 & 2 \\ 1 & 4 \\ 1 & 1 \\ 2 & 3 \\ 1 & 3 \end{bmatrix}, \ \boldsymbol{A}_2 = \begin{bmatrix} 3 & 3 \\ 2 & 1 \\ 3 & 1 \\ 3 & 2 \\ 2 & 0 \end{bmatrix}, \ \boldsymbol{A}_3 = \begin{bmatrix} 2 & 1 \\ 3 & 4 \\ 1 & 3 \\ 0 & 4 \\ 3 & 1 \end{bmatrix}. \tag{9}$$

100 samples are generated in each of three classes. In addition, a query belonging to each class is prepared. The number of detected samples K is fixed at 20.

In this example, the number of principal components is $R = 2$, the threshold is $\gamma = 1 - 10^{-4}$, the parameters are $\lambda = 0$ and $\delta = 0.9999$. Sample generation and sample detection by the NN method and the NC method are repeated 100 times and the average discrimination rates [%] and the average CPU time [ms] are calculated. The computer configuration used in this numerical example is as follows: OS: Windows Vista Business (64bit), CPU: Intel Core2 Duo 6300 (1.86GHz×2), RAM: 2G byte, and MATLAB® 7.5.0 (2008a).

Table 1 shows the discrimination results of the NN method and the NC method. The proposed NC method can achieve higher discrimination performance than the NN method. On the other hand, the computational load of the NC method is relatively heavy since singular value decomposition (SVD) is used for calculating the correlation among variables. In fact, the computation of SVD occupies most of the computation time of the NC method.

4. CORRELATION-BASED JUST-IN-TIME MODELING

The conventional JIT modeling uses the distance for sample selection when a temporary local model is constructed. However, its estimation performance is not always high since it does not take into account the correlation among variables. Recently, The Correlation-based JIT (CoJIT) modeling that selects samples for local modeling on the basis of the correlation among variables has been proposed by Fujiwara et al. (2008).

Figure 3 shows the difference of sample selection for local modeling between the JIT modeling and the CoJIT modeling. The samples are classified into two groups that have

Table 1. Discrimination performance of the NC method and the NN method

	Discrimination rate [%]			CPU time [ms]
	Class 1	Class 2	Class 3	
NC method	97.5	95.9	96.9	13.9
NN method	78.7	68.0	51.1	1.1

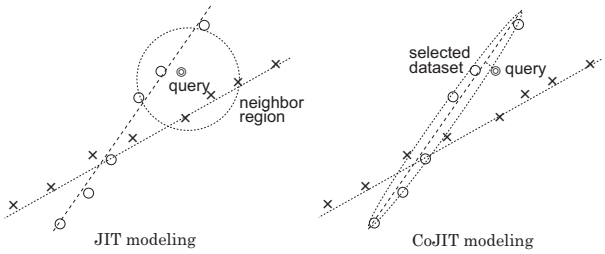


Fig. 3. Sample selection in the JIT modeling (left) and the CoJIT modeling (right)

different correlations. In conventional JIT modeling, samples are selected regardless of the difference of correlation as shown in Fig. 3 (left), since a neighbor region around the query point is defined only by distance. On the other hand, the CoJIT modeling can select samples whose correlation is best fit for the query as shown in Fig. 3 (right).

The procedure of the CoJIT modeling is as follows: 1) several data sets are generated from data stored in the database. 2) The index J is calculated from the query and each data set. 3) The data set whose J is the smallest is selected. 4) A temporary local model is constructed from the selected data set. In the above procedure, each data set is generated so that it consists of successive samples included in a certain period of time, because the correlation in such a data set is expected to be very similar (Fujiwara et al. (2008)).

However, the NC method can detect samples that have correlation similar to the query regardless of whether the objective data is time-series data or not. This is the motivation for integrating the proposed NC method with the CoJIT modeling.

Assume that the sampling interval of the output is longer than that of the input, and the output at time t, $\boldsymbol{y}_t$, should be estimated. Now, the input and the output measured at the same time are stored in the database, and the sth input-output sample $\boldsymbol{x}^{\{s\}} \in \Re^M$ ($s = 1, 2, \cdots, S$) and $\boldsymbol{y}^{\{s\}} \in \Re^L$ are stored as matrices $\boldsymbol{X}_S \in \Re^{S \times M}$ and $\boldsymbol{Y}_S \in \Re^{S \times L}$, respectively. To cope with process dynamics, measurements at different sampling times can be included in $\boldsymbol{x}^{\{s\}}$. The algorithm of the proposed CoJIT modeling with the NC method is as follows:

(1) When the input at time t, $\boldsymbol{x}_t$, is measured, the index J is calculated from $\boldsymbol{x}_t$ and $\boldsymbol{X}_{t-1}$ that was used for building the previous local model f_{t-1}, and $J_I = J$.
(2) If $J_I \leq \bar{J}_I$, $f_t = f_{t-1}$, $\boldsymbol{X}_t = \boldsymbol{X}_{t-1}$, and f_t is used for estimating the output $\boldsymbol{y}_t$. Then, return to step 1. If $J_I > \bar{J}_I$, go to the next step. Here, $\bar{J}_I$ is the threshold.
(3) K input samples whose correlation is similar to the query are detected from $\boldsymbol{X}_S$ by the NC method, and they are arranged as the rows of $\boldsymbol{X}_t \in \Re^{K \times M}$. In addition, K output samples corresponding to the detected input samples are selected from $\boldsymbol{Y}_S$, and they are arranged as the rows of $\boldsymbol{Y}_t \in \Re^{K \times L}$, where K is the number of the detected samples.
(4) A new local model f_t whose input is $\boldsymbol{X}_t$ and output is $\boldsymbol{Y}_t$ is built.
(5) The output $\boldsymbol{y}_t$ is estimated by using f_t.

(6) The above steps 1 through 5 are repeated until the next output sample $\boldsymbol{y}_{S+1}$ is measured. When $\boldsymbol{y}_{S+1}$ is measured, $\boldsymbol{y}_{S+1}$ and its corresponding input $\boldsymbol{x}_{S+1}$ are stored in the database, and return to step 1.

In the above algorithm, any modeling method can be used for building a local model f. In the present work, partial least squares regression (PLS) is used to cope with the colinearity problem. In addition, steps 1 and 2 control the model update frequency. When the threshold $\bar{J}_I$ is large, the update frequency becomes low. The local model is updated every time when new input measurements are available in the case where $\bar{J}_I = 0$.

5. CASE STUDY

In this section, the estimation performance of the proposed CoJIT modeling with the NC method is compared with that of the conventional JIT modeling through their applications to product composition estimation for a parallelized CSTR process. The detailed CSTR model used in this case study is described in Johannesmeyer and Seborg (1999).

5.1 Problem setting

In this process, CSTR1 and CSTR2 are operated in parallel. Although these CSTRs have the same structure

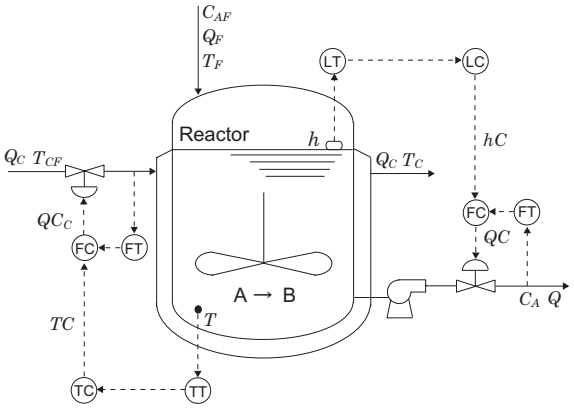


Fig. 4. Schematic diagram of CSTR with cascade control systems

Table 2. Process variables of the CSTR processes

Variable	Caption
C_A	Reactant concentration [mol/m^3]
T	Reactor temperature [K]
T_C	Coolant temperature [K]
h	Reactor level [m]
Q	Reactor exit flow rate [m^3/min]
Q_C	Coolant flow rate [m^3/min]
Q_F	Reactor feed flow rate [m^3/min]
C_{AF}	Feed concentration [mol/m^3]
T_F	Feed temperature [K]
T_{CF}	Coolant feed temperature [K]
hC	Level controller instruction
QC	Outlet flow rate controller instruction
TC	Temperature controller instruction
QC_C	Colorant flow rate controller instruction
T_{set}	Reactor temperature set point [K]

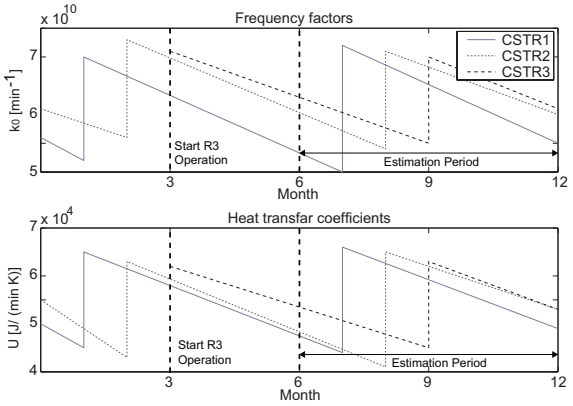


Fig. 5. Changes of overall heat transfer coefficients and frequency factors of the CSTRs

as shown in Fig. 4, they have different characteristics. In each CSTR, an irreversible reaction A $\longrightarrow$ B takes place. The set point of the reactor temperature $T^{[d]}(d = 1, 2)$ is independently changed between $\pm$2K every ten days. Although 15 process variables listed in Table 2 are calculated in the simulations, measurements of only five variables $T^{[d]}$, $h^{[d]}$, $Q^{[d]}$, $Q_C^{[d]}$, $Q_F^{[d]}$ are used for analysis, and their sampling interval is one minute. In addition, reactant concentration $C_A^{[d]}$ is measured in a laboratory once a day.

In this case study, to take into account catalyst deactivation and fouling as changes in process characteristics and individuality of each CSTR, the frequency factor $k_0^{[d]}$ and the heat transfer coefficient $UAc^{[d]}$ are assumed to decrease with time. In addition, each CSTR is maintained every half year (180 days). Figure 5 shows changes of the frequency factors $k_0^{[d]}$ and heat transfer coefficients $UAc^{[d]}$. The operation data of each CSTR for the half years (180 days) were stored in the database.

The soft-sensor for estimating reactant concentration of the newly developed CSTR3 is designed. The estimation of CSTR3 starts the 90th day after the start of its operation, and the soft-sensor is updated in the next half year. Although CSTR3 has only a small amount of data due to its short operation term, the soft-sensor is updated searching samples similar to the current operation of CSTR3 from the other CSTR operation data in the past.

5.2 Estimation result

The reactant concentration $C_A^{[3]}$ is estimated by the JIT modeling and the proposed CoJIT modeling with the NC method. To take into account process dynamics, the input data consist of the present sample and the sample measured one minute before.

In the JIT modeling, linear local models are built and Euclidean distance is used as the measure for selecting samples to build local models. The MATLAB Lazy Learning Toolbox developed by Bontempi et al. (1999) is used.

In the CoJIT modeling, samples for local modeling are selected by the NC method, and PLS is used for model building. The parameters of the NC method are deter-

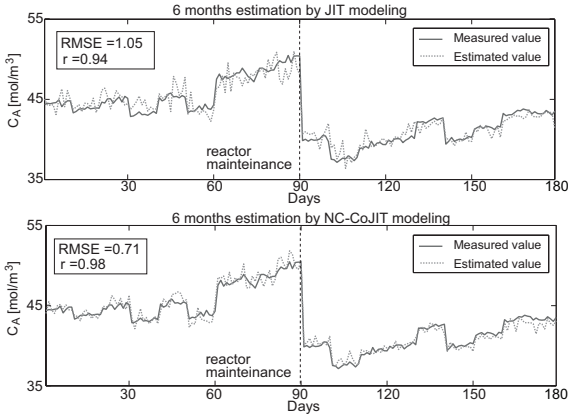


Fig. 6. Prediction result of $C_A^{\{3\}}$ by the JIT modeling (top) and the CoJIT modeling (bottom)

mined by trial and error, the threshold is $\gamma = 1 - 10^{4}$, the parameter is $\lambda = 0.01$, and the parameter for update frequency $\bar{J}_I = 0$.

The soft-sensor design results are shown in Fig. 6. Although $C_A^{[3]}$ is estimated every minute, only estimates corresponding to the measurements are plotted. In this figure, r denotes the correlation coefficient between measurements and estimates, and RMSE is the root-mean-squares error.

This result shows that the JIT modeling does not function well. On the other hand, the estimation performance of the proposed CoJIT modeling with the NC method is very high. With the proposed CoJIT modeling, RMSE is improved by about 35% in comparison with the JIT modeling. These results of this case study clearly show that the proposed CoJIT modeling can cope with not only abrupt changes in process characteristics but also the individuality of production devices. In addition, it can construct a high performance soft-sensor for a newly develop device, even if only a small amount of operation data is available.

6. CONCLUSION

A new unsupervised pattern recognition method that can detect samples whose correlation is similar to the query is proposed. In addition, the JIT modeling is integrated with the proposed the NC method. The proposed CoJIT modeling with the NC method can cope with not only changes in process characteristics but also the individuality of production devices and improve the estimation performance of a soft-sensor since it can select samples for local modeling by appropriately accounting for the correlation among variables. The proposed CoJIT modeling has the potential for realizing efficient maintenance of soft-sensors.

REFERENCES

Atkeson, CG., Moore, AW., and Schaal, S. (1997). Locally Weighted Learning, Artificial Intelligence Review, 11, 11-73.

Amirthalingam, R., and Lee J. (1999). Subspace Identification Based Inferential Control Applied to a Continuous Pulp Digester. J Proc Cont, 9,397-406.

Bontempi, G., Birattari, M., and Bersini, H. (1999). Lazy Learning for Local Modeling and Control Design. Int J Cont, 72, 643-658.

Bontempi, G., Birattari, M., and Bersini, H. (1999). Lazy Learners at Work: The Lazy Learning Toolbox. EUFIT'99: The 7th European Congress on Intelligent Techniques and Soft Computing, Aachen, Germany. Sep.13-16.

Cheng C., and Chiu, MS. (2004). A New Data-Based Methodology for Nonlinear Process Modeling. Chem Engng Sci, 59, 2801-2810.

Fujiwara, K., Kano, M., and Hasebe, S. (Accepted). Soft-Sensor Development Using Correlation-Based Just-In-Time Modeling. AIChE J.

Jackson, JE., and Mudholkar, GS. (1979). Control Procedures for Residuals Associated with Principal Component Analysis. Technometrics, 21, 341-349.

Johannesmeyer, M., and Seborg, DE. (1999). Abnormal Situation Analysis Using Pattern Recognition Techniques and Histrical Data. AIChE Annual meeting, Dallas, TX, Oct.31-Nov.5.

Kamohara, H., Takinami. A., Takeda. M., Kano, M., Hasebe, S., and Hashimoto, I. (2004). Product Quality Estimation and Operating Condition Monitoring for Industrial Ethylene Fractionator. J Chem Eng Japan, 37, 422-428.

Kano, M., Lee, S., and Hasebe, S. (2008). Two-Stage Subspace Identification for Softsensor Design and Disturbance Estimation. J Proc Cont, 1016/j.jprocont.2008.04.004.

Kano, M., Miyazaki, K., Hasebe, S., and Hashimoto, I. (2000). Inferential Control System of Distillation Compositions Using Dynamic Partial Least Squares Regression. J Proc Cont, 10, 157-166.

Kano, M., and Nakagawa, Y. (2008). Data-Based Process Monitoring, Process Control, and Quality Improvement. Recent Developments and Applications in Steel Industry, Comput Chem Engng, 32, 12-24.

Kohonen, T. (2001). Self-organizing maps. New York, Springer, 3rd edition.

Kresta, VJ., Marlin, TE., and MacGregor, JF. (1994). Development of Inferential Process Models Using PLS. Comput Chem Engng, 18, 597-611.

Mejdell, T., Skogestad, S. (1991). Estimation of Distillation Compositions from Multiple Temperature Measurements Using Partial-Least-Squares Regression. Ind Eng Chem Res, 30, 2543-2555.

Ogawa, M., and Kano, M. (2008). Practice and Challenges in Chemical Process Control Applications in Japan. The 17th IFAC World Congress, Paper WeC25.3.

Qin, SJ. (1998) . Recursive PLS Algorithms for Adaptive Data Modeling. Comput Chem Engng, 22, 503-514.

Raich, A. and Cinar, A. (1994). Statistical Process Monitoring and Disturbance Diagnosis in Multivariable Continuous Processes. AIChE J, 42, 995-1009.

Radhakrishnan, V., and Mohamed, A. (2000). Neural networks for the identification and control of blast furnace hot metal quality. J Proc Cont, 10,509-524.

Vesanto, J., Himberg, J., Alhoniemi, E., and Parhankangas, J. (1999). Self-Organizing Map in Matlab: the SOM Toolbox. The Matlab DSP Conference 1999, Espoo, Finland, Nov.16-17.

Process Monitoring

Oral Session

On-line statistical monitoring of batch processes using Gaussian mixture model

Tao Chen *, **Jie Zhang** **

* *School of Chemical and Biomedical Engineering, Nanyang Technological University, Singapore 637459 (e-mail: chentao@ntu.edu.sg).*
** *School of Chemical Engineering and Advanced Materials, Newcastle University, Newcastle upon Tyne, NE1 7RU, U.K. (e-mail: jie.zhang@ncl.ac.uk)*

Abstract: The statistical monitoring of batch manufacturing processes is considered. It is known that conventional monitoring approaches, e.g. principal component analysis (PCA), are not applicable when the normal operating conditions of the process cannot be sufficiently represented by a Gaussian distribution. To address this issue, Gaussian mixture model (GMM) has been proposed to estimate the probability density function of the process nominal data, with improved monitoring results having been reported for continuous processes. This paper extends the application of GMM to on-line monitoring of batch processes, and the proposed method is demonstrated through its application to a batch semiconductor etch process.

Keywords: Batch processes, mixture model, principal component analysis, probability density estimation, multivariate statistical process monitoring.

1. INTRODUCTION

Batch processing is of great importance in many industrial applications due to its flexibility for the production of low-volume, high-value added products. With increasing commercial competition it is crucial to ensure consistent and high product quality, as well as process safety. These requirements have resulted in wide acceptance of the technique of multivariate statistical process monitoring (Martin et al., 1999; Qin, 2003). The basis of the monitoring schemes is historical data that has been collected when the process is running under normal operating conditions (NOC). This data is then used to establish confidence bounds for the monitoring statistics, e.g. Hotelling's T^2 and squared prediction error (SPE), to detect the onset of process deviations. The primary objective of process monitoring is to identify abnormal behavior as early as possible, in addition to keeping an acceptably low false alarm rate.

As a result of the multi-way characteristic of batch process data, special tools are required for the modelling and monitoring purposes, including multi-way principal component analysis (MPCA) (Nomikos and MacGregor, 1995b), hierarchical PCA (Rannar et al., 1998) and multi-way partial least squares (MPLS) (Nomikos and MacGregor, 1995a). The methods for on-line monitoring of batch process can be classified into two categories. The first does not require measurements on the entire batch duration to be available. Techniques that are within this class include hierarchical and two-dimensional dynamic PCA (Rannar et al., 1998; Lu et al., 2005). In the other category, the entire batch data is required for the calculation of the monitoring statistics, whilst the data from a new batch is available only up to the current time. Therefore the future data must be predicted in some way (Nomikos and MacGregor, 1995b). In this paper the latter of the two approaches is considered, and the details will be discussed subsequently in Section 2.

However, the afore reviewed conventional process monitoring methods are based on a restrictive assumption that the NOC can be represented by a multivariate Gaussian distribution. Specifically the confidence bounds for T^2 and SPE are calculated by assuming the PCA/PLS scores and prediction errors are Gaussian distributed. This assumption may be invalid when the process data is collected from a complex manufacturing process. To address this issue, Gaussian mixture model (GMM) (Chen et al., 2006; Choi et al., 2004; Thissen et al., 2005), which is capable of approximating any probability density function (*pdf*), has been proposed for the monitoring of continuous processes, as well as batch-wise monitoring of batch processes.

The major contribution of this paper is to extend the application of GMM to on-line monitoring of batch processes. As the first step MPCA is applied to the nominal batch data to extract the low-dimensional representation of the process. The challenge with on-line monitoring is that the scores and SPE must be predicted based on available process measurements up to the current time step. Clearly the predicted scores and SPE are not identical to the values that are calculated from the entire batch duration, and thus the predictions may not conform to the nominal distribution even if the process is running normally. We follow the approach of Nomikos and MacGregor (1995b) to pass the nominal batches through the monitoring procedure and collect the predicted scores and SPE at each time step. Then GMM is employed to estimate the joint *pdf* of these predicted scores and SPE from MPCA at each

time step, as opposed to the traditional T^2 and SPE where the process data is assumed to be Gaussian distributed.

The rest of this paper is organized as follows. Section 2 gives a summary of the PCA and GMM tools for process monitoring, followed by the discussion of the on-line monitoring strategy in Section 3. Section 4 demonstrates the application of the on-line monitoring techniques to a batch semiconductor manufacturing process. Finally Section 5 concludes this paper.

2. PCA AND GAUSSIAN MIXTURE MODEL FOR PROCESS MONITORING

This section presents a brief overview of the PCA and GMM techniques. A number of issues related to the application to process monitoring are discussed, including model selection and the construction of confidence bound.

2.1 PCA

Principal component analysis (PCA) (Jolliffe, 2002) is a general multivariate statistical projection technique for dimension reduction, where the original data is linearly projected onto low-dimensional space such that the variance is maximized. Formally the D-dimensional data $\mathbf{x}$ is represented by a linear combination of the Q-dimensional scores $\mathbf{t}$ plus a noise vector $\mathbf{e}$: $\mathbf{x} = \mathbf{W}\mathbf{t} + \mathbf{e}$, where $\mathbf{W}$ are the eigenvectors of the sample covariance matrix having the Q largest eigenvalues ($Q \leq D$). Consequently normal process behavior can be characterized by the first Q principal components, which capture the main source of data variability.

The proper number of principal components can be selected using a number of criteria, including variance ratio, cross-validation and the "broken-stick" rule (Jolliffe, 2002). This is essentially a model selection problem. The "broken-stick" rule is adopted in this paper due to its low computation and good results reported in the literature (Nomikos and MacGregor, 1995b). According to this rule, the q-th principal component should be retained if the percentage of variance explained by it exceeds the corresponding G value given by

$$G(q) = \frac{100}{C} \sum_{i=q}^{C} \frac{1}{i} \tag{1}$$

where $C = \min(D, N)$.

In statistical process monitoring, the next step is to define the monitoring statistics and the corresponding confidence bounds. Traditionally two metrics are used: $T^2 = \mathbf{t}^{\mathrm{T}}\mathbf{\Lambda}^{-1}\mathbf{t}$ and SPE as $r = \mathbf{e}^{\mathrm{T}}\mathbf{e}$, where $\mathbf{\Lambda}$ is a diagonal matrix comprising the Q largest eigenvalues.

As discussed previously, the first issue with T^2 and SPE is that the corresponding confidence bounds are calculated based on restrictive Gaussian distribution. Secondly two separate metrics are required for process monitoring. Practically the process is identified as deviating from normal operation if either T^2 or SPE moves outside the confidence bounds. This empirical solution could potentially increase the false alarm level [1]. The technique of GMM is suitable for addressing the two issues simultaneously. In our previous work (Chen et al., 2006) we have demonstrated that a unified monitoring statistic can be obtained by estimating the joint *pdf* of the PCA scores and log-SPE using GMM, i.e. the *pdf* of a $(Q+1)$-dimensional vector $\mathbf{z} = (\mathbf{t}^{\mathrm{T}}, \log r)^{\mathrm{T}}$. The logarithm operator is used to transform the non-negative SPE onto the whole real axis on which the GMM is defined.

In this paper the methodology in (Chen et al., 2006) is followed to establish the confidence bounds for process monitoring based on PCA and GMM techniques. GMM is described in detail in the next subsection.

2.2 Gaussian mixture model

As a general tool for *pdf* estimation, Gaussian mixture model (GMM) has been used in a wide variety of problems in applied statistics and pattern recognition. A GMM is a weighted sum of M component densities, each being a multivariate Gaussian with mean $\boldsymbol{\mu}_i$ and covariance matrix $\mathbf{\Sigma}_i$:

$$p(\mathbf{z}|\boldsymbol{\theta}) = \sum_{i=1}^{M} \alpha_i G(\mathbf{z};\, \boldsymbol{\mu}_i, \mathbf{\Sigma}_i) \tag{2}$$

where the weights satisfy the constraint: $\sum_{i=1}^{M} \alpha_i = 1$. A GMM is parameterized by the mean vectors, covariance matrices and mixture weights: $\boldsymbol{\theta} = \{\alpha_i, \boldsymbol{\mu}_i, \mathbf{\Sigma}_i; i = 1, \ldots, M\}$.

Given a set of training data $\{\mathbf{z}_n, n = 1, \ldots, N\}$, the parameters can be estimated by maximizing the likelihood function: $L(\boldsymbol{\theta}) = \prod_{n=1}^{N} p(\mathbf{z}_n|\boldsymbol{\theta})$. In the context of process monitoring, $\mathbf{z}_n$ is the $(Q+1)$-dimensional vector of PCA scores and log-SPE: $\mathbf{z}_n = (\mathbf{t}_n^{\mathrm{T}}, \log r_n)^{\mathrm{T}}$. The maximization is typically implemented iteratively using the expectation-maximization (EM) algorithm (Dempster et al., 1977).

The number of mixture components, M, must be selected prior to the training of a GMM. This is a model selection problem that can be addressed using a number of methods, including cross-validation and Bayesian information criterion (BIC) (Schwarz, 1978). BIC is widely applied in model selection problems for its effectiveness and low computational cost. According to BIC the model is selected such that $L - (H/2)\log N$ is the largest, where L is the log-likelihood of the data and H is the total number of parameters within the model. The motivation of BIC is that a good model should be able to sufficiently explain the data (the log-likelihood) with low model complexity (the number of parameters). In this study BIC is adopted for the selection of number of mixtures.

One of the advantages of the GMM for process monitoring is that it provides the likelihood value as the single statistic for the construction of confidence bounds, as opposed to the confidence bounds for two statistics (i.e. the T^2 and SPE) in conventional process monitoring techniques. In

[1] Suppose 95% confidence bound is used, and thus by definition the false alarm rate is 5% for both T^2 and SPE. The probability of either T^2's bound or SPE's bound being exceeded, when the process is runing normally, will be equal to or greater than 5%.

practice a single monitoring statistic simplifies the plant operators' decision effort, and it may be more sensitive to some subtle process faults (Chen et al., 2006).

On the basis of the *pdf* $p(\mathbf{z}|\boldsymbol{\theta})$ for the normal operating data, the $100\beta\%$ confidence bound is defined as a likelihood threshold h that satisfies the following integral (Chen et al., 2006):

$$\int_{\mathbf{z}:p(\mathbf{z}|\boldsymbol{\theta})>h} p(\mathbf{z}|\boldsymbol{\theta})d\mathbf{z} = \beta \tag{3}$$

To determine the confidence bound, we can calculate the likelihood of all the nominal data, and then find h that is less than the likelihood of $100\beta\%$ (e.g. 99%) of the nominal data (Thissen et al., 2005). This approach is applicable to most continuous processes where the number of nominal data points can be up to several thousand; however it may be unreliable when the nominal data is very limited as in batch process monitoring. The estimation of the confidence bound based on limited batches would be very sensitive to the data, and thus a small perturbation in the data would result in very different estimation of the h.

To address this issue, we resort to numerical Monte Carlo simulation to approximate the integral in Eq. (3) (Chen et al., 2006). Specifically we generate N_s random samples, $\{\mathbf{z}^j, j = 1, \ldots, N_s\}$, from $p(\mathbf{z}|\boldsymbol{\theta})$. These samples serve as the "pseudo data" (since the real data is not sufficient) to represent the normal process behavior. Thus the Monte Carlo samples, in conjunction with nominal process data, are used to calculate the confidence bound h. Then a new batch $\mathbf{z}$ is considered to be faulty if $p(\mathbf{z}|\boldsymbol{\theta}) < h$ (or equivalently $-p(\mathbf{z}|\boldsymbol{\theta}) > -h$). The number of Monte Carlo samples required (N_s) to approximate the confidence bounds is dependent on the dimension of $\mathbf{z}$, and it can be determined heuristically.

3. MONITORING OF BATCH PROCESSES

To analyze the three-way batch data ($N \times J \times K$) (N, J and K denote the number of batches, process variables at each time instance, and time steps, respectively), multiway analysis methods have been proposed to unfold the data array into a two-way matrix on which conventional PCA is then performed (Nomikos and MacGregor, 1995b). This study unfolds the data array into a large matrix ($N \times JK$) such that each batch is treated as a "data point". This two-way matrix is then pre-processed to zero mean and unit standard deviation on each column, prior to the application of PCA to extract the scores $\mathbf{t}_n$ and SPE r_n, $n = 1, \ldots, N$. Then a Gaussian mixture model is developed for the joint vector $\mathbf{z}_n = (\mathbf{t}_n^{\mathrm{T}}, \log r_n)^{\mathrm{T}}$, followed by the calculation of confidence bound using Monte Carlo simulation.

3.1 On-line monitoring

In the on-line monitoring stage, it is necessary to project the new batch onto the PCA space to obtain the scores and SPE, and then to calculate the likelihood value under the GMM to identify possible process anomaly. The issue is that, at time step t, the batch measurements are only available up to the current time. It is possible to develop multiple PCA and GMM models at each time step; however this strategy requires excessive computation and computer memory. A more reasonable and widely accepted method is to predict the scores and SPE using the available measurements.

More specifically, let $\bar{\mathbf{x}}_{1:t}$ be the vector of a new batch with available measurements from time step 1 to t. Note $\bar{\mathbf{x}}_{1:t}$ is a vector of order Jt. According to Nomikos and MacGregor (1995b), the least square prediction of the scores is:

$$\bar{\mathbf{t}}_{1:t} = \left(\mathbf{W}_{1:t}^{\mathrm{T}}\mathbf{W}_{1:t}\right)^{-1}\mathbf{W}_{1:t}^{\mathrm{T}}\bar{\mathbf{x}}_{1:t} \tag{4}$$

where $\mathbf{W}_{1:t}$ is the sub-matrix of $\mathbf{W}$ having the rows corresponding to time step 1 to t. In Eq. (4) the matrix to be inverted is well conditioned due to the orthogonality of the loading $\mathbf{W}$. Since the future measurements are not available, the prediction error can only be calculated up to time step t:

$$\bar{\mathbf{e}}_{1:t} = \bar{\mathbf{x}}_{1:t} - \mathbf{W}_{1:t}\bar{\mathbf{t}}_{1:t} \tag{5}$$

The SPE is then obtained as $\bar{\mathbf{e}}_{1:t}^{\mathrm{T}}\bar{\mathbf{e}}_{1:t}$. It was suggested to use the "instantaneous" SPE associated with the latest on-line measurements for process monitoring (Nomikos and MacGregor, 1995b), i.e. $\bar{\mathbf{e}}_t^{\mathrm{T}}\bar{\mathbf{e}}_t$, which is expected to increase the sensitivity of fault detection method. However the instantaneous SPE leads to an excessive number of false alarms in the case study of this paper (see details in Section 4). This phenomenon could be due to the non-Gaussian distribution of the process data. The SPE calculated from Eq. (5), which in a sense is a smoothed version of the instantaneous SPE, may be a more appropriate monitoring metric. We will discuss this issue through the application study in Section 4.

Clearly the predicted scores and SPE from Eqs. (4)(5), based on current available measurements, are not identical to the values that are calculated should the entire batch be available. As a result the predicted scores and SPE may not conform to the *pdf* developed based on the entire duration of nominal batches, even if the process being monitored is running normally. This is a serious issue particularly in the initial stage of a batch processing, when only a small number of measurements are available to calculate the scores and SPE. We follow the standard approach in on-line batch process monitoring (Nomikos and MacGregor, 1995b) to pass each of the nominal batches through the monitoring procedure to collect the predicted scores and SPE at each time step from Eqs. (4)(5), and then apply GMM to estimate the joint *pdf* of these predicted scores and log-SPE at each time step, and to establish the confidence bounds as presented in Section 2. Essentially we propose to replace the confidence bounds for T^2 and SPE in (Nomikos and MacGregor, 1995b), where the process data is assumed to be Gaussian distributed, with more powerful Gaussian mixture model. For on-line monitoring of a new batch, the scores and SPE are calculated from Eqs. (4)(5), and the likelihood value is calculated under the GMM for the current time step. If this likelihood value is lower than the confidence bound, the process under monitoring is considered to be in a faulty condition.

Table 1. Variables used for the monitoring of the semiconductor process.

1	Endpoint A detector	7	RF impedance
2	Chamber pressure	8	TCP tuner
3	RF tuner	9	TCP phase error
4	RF load	10	TCP reflected power
5	RF Phase error	11	TCP Load
6	RF power	12	Vat valve

4. CASE STUDY

The manufacture of semiconductors is introduced as an example of the on-line monitoring of batch processes. This study focuses specifically on an Al-stack etch process performed on the commercially available Lam 9600 plasma etch tool (Wise et al., 1999). Data from 12 process sensors, listed in Table 1, was collected during the wafer processing stage which run for 80 s. A sampling interval of 1 s was used in the analysis. Thus for each batch, the data is of the order (12×80). A series of three experiments, resulting in three distinct data groups, were performed where faults were intentionally introduced by changing specific manipulated variables (TCP power, RF power, pressure, plasma flow rate and Helium chunk pressure). There are 107 normal operating batches and 20 faulty batches. Twenty batches were randomly selected from the normal batches to investigate the effect of false alarms. The remaining 87 nominal batches were used to build the MPCA and GMM models.

4.1 Off-line analysis

According to MPCA, the three-way nominal data array $(N \times J \times K = 87 \times 12 \times 80)$ is unfolded into a large two-way matrix of the order (87×960), which is then mean-centered and scaled to unit standard deviation on each column. Then PCA is applied to the pre-processed data, where two principal components are retained according to the broken-stick rule (Jolliffe, 2002). Considering there are 960 columns in the unfolded matrix, it is not surprising to find that two principal components explain only 45.50% of the total variance (similar results can be found in the literature, e.g. (Nomikos and MacGregor, 1995b)).

Figure 1 gives the scatter plot of the PCA scores corresponding to the first two principal components. It is clear that the nominal data exhibits the characteristic of multiple groups, and it cannot be adequately approximated by a single multivariate Gaussian distribution. As a result, the 99% confidence bound does not capture the region of NOC accurately. In addition to the normal testing batches, 17 out of 20 faulty batches are within the confidence bound, resulting in 17 missing errors. Clearly more complex models are required to represent the nominal behavior of the process.

To develop a GMM for the PCA scores and log-SPE, the appropriate number of mixtures must be selected. According to the BIC, the GMM with three mixture components is utilized for the analysis of the semiconductor process. Once the GMM is developed, the 95% and 99% confidence bounds is calculated using Monte Carlo simulation presented in Section 2.2, where the number of random samples is heuristically determined to be 10,000. Despite the large sample size, the CPU time for the Monte Carlo simulation

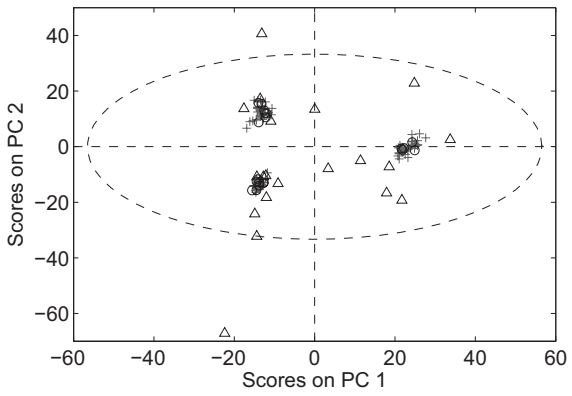


Fig. 1. Bivariate scores plot for principal components 1 and 2 with 99% confidence bound (− − −−): nominal (+), normal (∘) and faulty (△).

Table 2. Off-line monitoring results.

	T^2	SPE	T^2 + SPE	GMM
False alarms	0	0	0	0
Missing errors	17	7	7	4

was only 0.03 s (Matlab implementation under Windows XP with Pentium 2.8 GHz CPU). In the literature the 95% is treated as "warning bound" and 99% "action bound". Throughout this section the process is classified as faulty if the 99% confidence bound is violated.

Table 2 summarizes the off-line batch-wise monitoring results for both conventional PCA and the GMM approach. Both methods incur no false alarms in this example. The large number of missing errors from T^2, as depicted in Figure 1, is the result of over-estimation of the confidence bound. It appears that SPE is more sensitive to the fault and it attains seven missing errors. By combining T^2 and SPE in the way that the process is identified as faulty if either metric is exceeded, the number of missing errors is still seven. Table 2 clearly indicates that GMM outperforms the conventional PCA in terms of smaller number of missing errors through the direct estimation of the joint *pdf* of the PCA scores and log-SPE.

4.2 On-line monitoring

The on-line monitoring results are given in Table 3. A normal testing batch is considered to be a false alarm if it is identified as faulty within the batch duration. A missing error means a faulty batch is not detected during the entire duration. Similar to the off-line monitoring, T^2 fails to detect most of the faulty batches because the scores do not conform to a multivariate Gaussian distribution. A comparison between Table 3 (a) and (b) suggests that the instantaneous SPE can detect more faulty batches than the smoothed SPE; however the increased sensitivity is at the cost of dramatically decreased robustness. The number of false alarms for instantaneous SPE is excessively large (13 out of total 20 batches), and thus the smoothed SPE is adopted for the rest of this paper. Table 3 indicates that the GMM approach gives better results than the conventional MPCA in terms of smaller number of false alarms and missing errors.

Table 3. On-line monitoring results. (a) SPE is calculated based on process measurements at current time step (instantaneous SPE); (b) SPE is calculated based on process measurements from batch beginning to current time step (smoothed SPE).

(a)

	T^2	SPE	T^2 + SPE	GMM
False alarms	0	13	13	8
Missing errors	17	2	2	0

(b)

	T^2	SPE	T^2 + SPE	GMM
False alarms	0	3	3	1
Missing errors	17	4	4	2

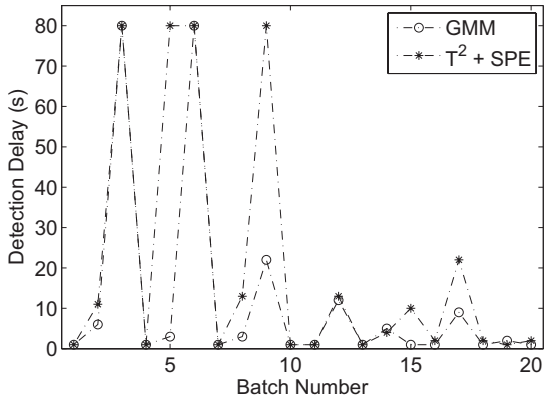


Fig. 2. Delay in the detection of the faulty batches.

It should be noted that the number of missing errors in on-line monitoring is not the only index to evaluate the monitoring performance. Of greater practical importance is the time delay between the occurrence and the detection of the fault. Figure 2 illustrates the detection delay of the 20 faulty batches using MPCA and GMM. To facilitate the calculation of average delay for comparison, the detection delay is artificially set to the batch duration (i.e. 80 s) if a faulty batch is not detected by the monitoring system. Essentially this is to assume that the abnormal behavior will be identified in some way (e.g. the presence of off-specified product) when the batch finishes. In practice plant operators are often not able to identify the fault until much later than the end of batch duration. On average, the detection delay for GMM is 11.6 s that is significantly shorter than 20.3 s obtained by the PCA method. Since the process is operating relatively fast, the reduction of delay in 9 s (equivalently 9 time steps) may not be sufficient for the operators to take appropriate actions in practice. Nevertheless if the proposed approach is applied to monitor a slow process, for example batch fermentation that takes several days to complete where data is sampled every half day, a shorter detection delay of 9 time steps would provide significant advantage in terms of reduced operational cost and improved process safety and product quality.

Figure 3 illustrates the on-line monitoring charts of a normal batch, which is false-alarmed by conventional PCA. Since the value of on-line SPE increases with time, we

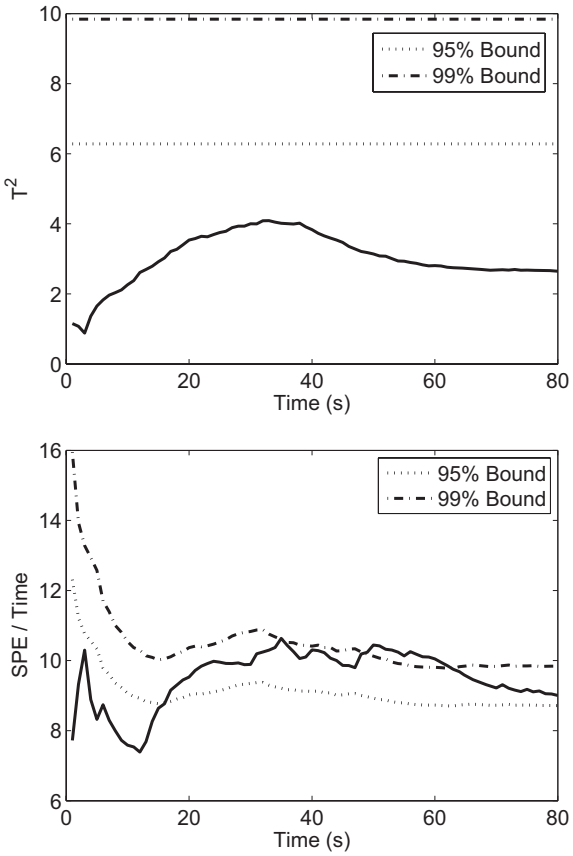


Fig. 3. On-line monitoring of a normal batch using T^2 and SPE.

plot SPE divided by time for better illustration in the figure. The T^2 indicates that this batch is under normal operation; however T^2 is not a reliable index for the monitoring of this process as discussed previously. The SPE metric appears to be susceptible to process disturbance; it exceeds the 95% confidence bound from 17 s and is over the 99% bound between 50 s to 60 s, despite the fact that the process is running normally. Figure 4 shows the GMM based monitoring chart, where the negative likelihood value is plotted. The GMM approach correctly recognizes that this batch is within the region of NOC during the whole batch duration.

Figure 5 and 6 give the on-line monitoring charts of a faulty batch (batch 5 as in Figure 2), using conventional PCA and the GMM approach, respectively. Both T^2 and SPE fails to detect this fault. In contrast, the likelihood value from the GMM is becoming outside the 99% confidence bound since time 3 s.

5. CONCLUSIONS

This paper extends the GMM technique for the modelling and on-line performance monitoring of batch manufacturing processes. The handling of the unobserved future batch measurements is discussed for the purpose of on-line monitoring. The GMM provides a probabilistic approach to estimating the *pdf* of the nominal process data and therefore enables more accurate calculation of the

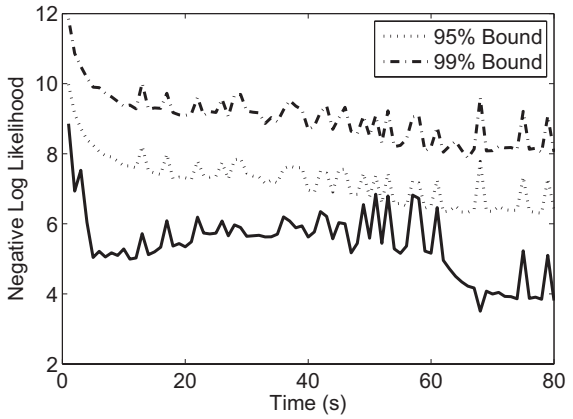


Fig. 4. On-line monitoring of a normal batch using GMM.

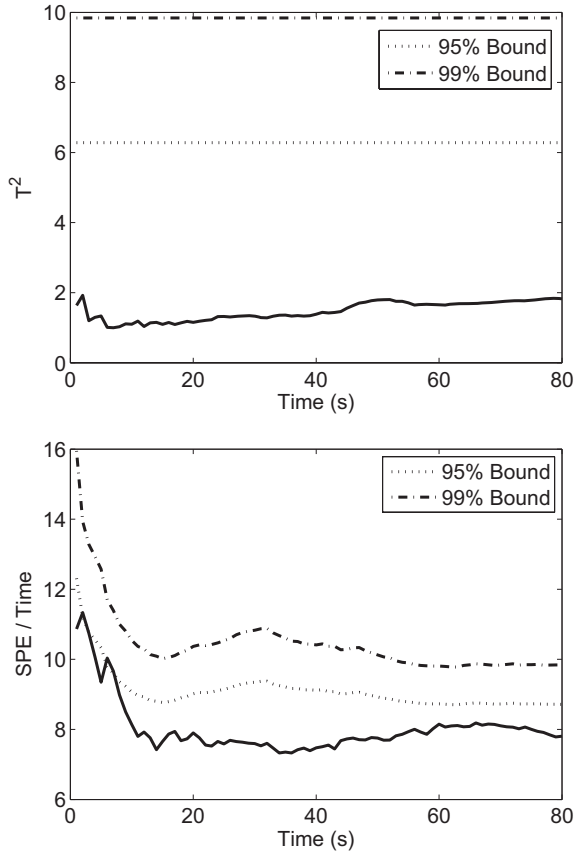


Fig. 5. On-line monitoring of a faulty batch using T^2 and SPE.

confidence bounds. The case study confirms that through accurate modelling of the process historical data collected from NOC, GMM is a promising approach to maintaining a low rate of both false alarms and missing errors in process performance monitoring.

REFERENCES

Chen, T., Morris, J., and Martin, E. (2006). Probability density estimation via an infinite Gaussian mixture model: application to statistical process monitoring. *Journal of the Royal Statistical Society C (Applied Statistics)*, 55, 699–715.

Choi, S.W., Park, J.H., and Lee, I.B. (2004). Process monitoring using a Gaussian mixture model via principal component analysis and discriminant analysis. *Computers and Chemical Engineering*, 28, 1377–1387.

Dempster, A.P., Laird, N.M., and Rubin, D.B. (1977). Maximum likelihood from incomplete data via the EM algorithm. *Journal of Royal Statistical Society B*, 39, 1–38.

Jolliffe, I.T. (2002). *Principal Component Analysis*. Springer, 2nd edition.

Lu, N., Yao, Y., and Gao, F. (2005). Two-dimensional dynamic pca for batch process monitoring. *AIChE Journal*, 51, 3300–3304.

Martin, E.B., Morris, A.J., and Kiparrisides, C. (1999). Manufacturing performance enhancement through multivariate statistical process control. *Annual Reviews in Control*, 23, 35–44.

Nomikos, P. and MacGregor, J.F. (1995a). Multi-way partial least squares in monitoring batch processes. *Chemometrics and Intelligent Laboratory Systems*, 30, 97–108.

Nomikos, P. and MacGregor, J.F. (1995b). Multivariate SPC charts for monitoring batch processes. *Technometrics*, 37, 41–59.

Qin, S.J. (2003). Statistical process monitoring: basics and beyond. *Journal of Chemometrics*, 17, 480–502.

Rannar, H., MacGregor, J.F., and Wold, S. (1998). Adaptive batch monitoring using hierarchical PCA. *Chemometrics and Intelligent Laboratory Systems*, 41, 73–81.

Schwarz, G. (1978). Estimating the dimension of a model. *Annals of Statistics*, 6, 461–464.

Thissen, U., Swierenga, H., de Weijer, A.P., Wehrens, R., Melssen, W.J., and Buydens, L.M.C. (2005). Multivariate statistical process control using mixture modelling. *Journal of Chemometrics*, 19, 23–31.

Wise, B.M., Gallagher, N.B., Butler, S.W., White, D.D., and Barna, G.G. (1999). A comparison of principal component analysis, multiway principal component analysis, trilinear decomposition and parallel factor analysis for fault detection in a semiconductor etch process. *Journal of Chemometrics*, 13, 379–396.

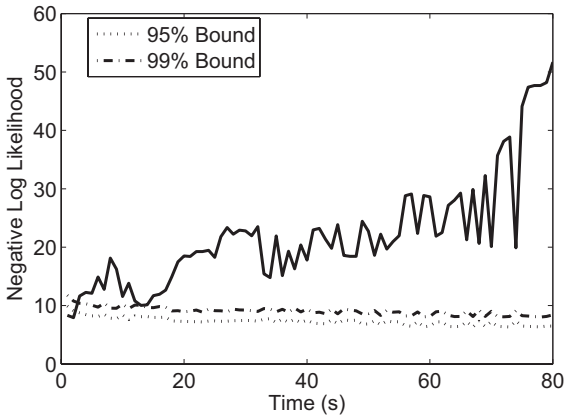


Fig. 6. On-line monitoring of a faulty batch using GMM.

Variability Matrix: A Novel Tool to Prioritize Loop Maintenance

Marcelo Farenzena*, Jorge O. Trierweiler*, Sirish L. Shah**

** Department of Chemical Engineering, Federal University of Rio Grande do Sul (UFRGS)*
Porto Alegre, RS, CEP: 90.040-040, BRAZIL (Tel:+555133084072; e-mail:{farenz,jorge}@enq.ufrgs.br)
*** Department of Chemical and Materials Engineering, University of Alberta, Edmonton, AB, T6G 2G6, Canada*
(e-mail: sirish.shah@ualberta.ca)

Abstract: It is now common knowledge that as many as 40% of the control loops in most industrial processes have considerable potential for improving control performance by reducing variability. Because of the large number of control loops in an industrial plant, controller performance monitoring is indispensable, but equally important is how to prioritize their maintenance. It is well known that variance reduction in a loop occurs by transferring variability to other variables or loops. The focus of this study is to propose a methodology to prioritize loop maintenance based on the potential improvement of each loop and the variability transfer among them. The central point of this work is the Variability Matrix (VM), an array that shows the impact of performance improvement of a given loop on the whole plant. Based on the VM, a methodology to translate this array into a potential loop economic benefit metric is also introduced. The VM can be quantified in the ideal scenario where plant model and controller are available and also when they are not, thus allowing the application of these ideas in industry. The efficacy of proposed methodology is illustrated by successful application to two case studies.

1. INTRODUCTION

The main requirement for a control system is to ensure process stability and robustness. This is the key reason for the widespread industrial interest in performance assessment methodologies and tools. A typical plant has hundreds or thousands of controllers and most of them have potential for improvement (Bialkowski, 1993). Many good reviews on assessment of control loops are available in the literature (Huang and Shah, 1999, Jelali, 2006). A common problem in controller performance monitoring is how to prioritize loop maintenance. The answer should not only be based on the performance potential, but also on the economic benefits that can be realized in improving the performance of each loop.

The main motivation for improving the performance of the plant is simple: reduction in process variability allows achieving a more profitable operating point, closer to the constraints, as shown in Fig. 1. In scenario I, the process has large variability and therefore the setpoint or the target has to be significantly far away from the economically optimal operating point. If the variability is reduced, due to controller or process improvement (scenario II) the process operating point can be moved to a more profitable setpoint (scenario III).

The literature is relatively sparse in terms of quantification of economic benefits due to improvement of controller performance. Muske (2003) proposed the idea of potential reduction in control loop variability. The economic benefit is quantified based on the shift in the mean operation toward a product specification or process constraint. The variance reduction can be based on a fixed or user-specified benchmark, e.g. minimum variance benchmark or a desired

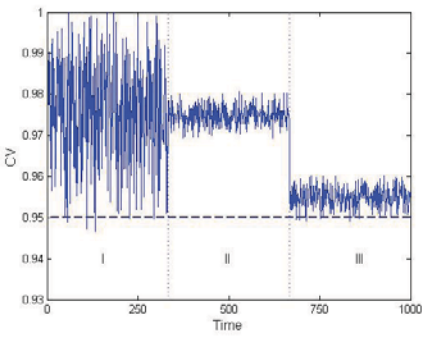


Fig. 1: Variability reduction impact: (I) normal operating variability (II) variability reduction and (III) operating point shift.

rise time or settling time benchmark. Craig and Henning (2000) proposed another methodology to quantify the economic benefit of Advanced Process Control (APC) projects. The authors mention that the whole part of the benefits come from the steady-state optimization. They assume that the variance of the products can be reduced by 35% to 50%. Mascio and Barton (2001) proposed a methodology to quantify the control quality in economic terms based on the Taguchi Framework.

All available methodologies agree that reduction in variability means shifting the operating point to a more profitable point. The main drawback is that they consider each loop as an isolated case, i.e. if performance of one loop is improved then the whole plant will not suffer its effect.

All modern industrial plants have significant interaction among loops due to tighter heat integration. Because of this,

one cannot assume that the variance reduction in one loop will occur without impacting other loops adversely. Typically, variability is transferred from loops where it should be reduced to loops that have the room or the buffer to accommodate large fluctuations (e.g. level loops). In many cases, if one variable has its variability reduced and its operating point shifted, then it is likely that other interacting or complementary loops will have their variability increased, shifting the operating point away from the constraints. This implies that "part of the profit" realized by variability reduction in a given loop "will be offset" by the loops where the variability increases. This is why a control loop should not be considered in isolation and the potential economic benefit should be computed by analyzing the whole plant and not only a specific loop. The common idea that the improvement of a given controller performance will increase the performance of the whole plant is not always true. Sometimes in an interacting system, the coupling between the channels can help or hinder overall performance. For example, decrease in the variability of a given controller can also reduce the variability in other loops in which case, one can say that the interaction helps. In other cases, the interaction may affect performance of associated loops adversely.

The main contribution of this work is the introduction of the notion of the Variability Matrix (VM). This array shows how the variability will transfer between the loops and the impact of one specific loop on the variances of all other interacting or complementary loops. The potential economic benefit of each loop can be quantified based on VM.

This paper is structured as follows: section 2 introduces the concept of Variability Matrix. In section 3, practical issues in computing the VM are discussed. The methodology to quantify the economic benefit of each control loop and prioritize loop maintenance is shown in section 4. The complete methodology is illustrated by successful application on two case studies (section 5). The paper ends with concluding remarks.

2. VARIABILITY MATRIX: CONCEPTS AND DEFINITION

2.1 Preliminary Definitions

To quantify the economic impact, it is interesting to consider the classification of control loops into the following two categories:

Main Loops: Loops that directly control the products specification. Their performance improvement affects the product variability, which can be directly translated into profitability.

Auxiliary Loops: Loops that do not directly control product quality, but can indirectly affect the product variability.

2.2 Variability Matrix Structure

The structure of the variability matrix consists of the following:

Rows: The rows show the influence of each loop on the same final product. The number of rows is the same as the products or the number of main loops.

Columns: Shows the influence of a specific loop on all other loops that may impact or influence the specification of the final product. The number of columns is the same as the number of control loops implemented in the plant. The first columns correspond to the main loops and the adjacent set of columns corresponds to the auxiliary loops as shown in Figure 2.

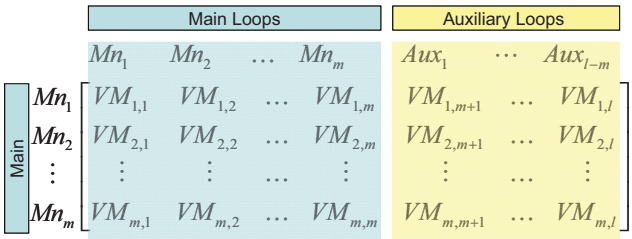


Fig. 2: Schematic representation of Variability Matrix

In Fig. 2 Mn_i is the main loop i and Aux_j is the auxiliary loop j. The total number of loops in the plant is l and it has m main loops. For example, column 1 (Mn_1) shows the impact of variability reduction in main controller 1 on all other main loops. Row 1 shows the impact on the variability of Mn_1 when the performance of all other loops is changed.

2.3 VM Computation

This section discusses the methodology for computing each element *VM(i,j)* of the Variability Matrix. In the first scenario, the following assumptions are taken: (I) the plant model (G) is available; (II) the controller model (C) is also available; and (III) the controlled variables (y) and control outputs (u) are available. For the sake of simplicity, we consider that the setpoint is fixed and set to zero.

Based on the previous assumptions, the procedure to quantify the VM is described below:

1. Read process data y_j ($j = 1…l$) and u_j ($j = 1…l$) with all loops closed (with actual performance);
2. Select main and auxiliary loops;
3. Compute the actual variance for each main loop ($\text{var}_{act,i}$, $i = 1…m$);
4. For each loop j ($j = 1…l$)
    1. Calculate the best performance achievable (see section 3.2) for loop j;
    2. Apply the controller;
    3. Calculate the new variance for each main loop i ($var_{best,i,j}$, $i = 1…m$)
    4. Compute the elements of VM j^{th} column using eq. 1.

$$VM(i,j) = \frac{\text{var}_{act,i} - \text{var}_{best,i,j}}{\text{var}_{act,i}} \quad (1)$$

This structure for VM elements was chosen because for two main reasons: 1) it provides a direct measure of the

variability improvement potential for each loop; and 2) it is dimensionless, a fact that allows the comparison of the impact of two or more loops in the plant. For example, consider the VM of:

$$\begin{bmatrix} 0.3 & 0 & -1.2 \\ -0.7 & 0.9 & -1.5 \end{bmatrix} \quad (2)$$

Initially, we can verify that this plant has 2 main loops and one auxiliary loop. From this VM, by examining column 1, we can conclude that: if the performance of main controller 1 is improved, its variance will decrease 30%; however, it has a negative and strong impact on another loop: its variance will increase by 70%. Is this healthy for the process? Clearly the answer to this question depends on the economic impact of each main loop. In column 2, the main loop 2 has potential reduction in variability of 90%. This controller has no influence on the main loop 1 variance; furthermore improving the performance of the auxiliary loop (3rd column) will lead to variability increase in both main loops.

In complement with the VM, the concept of the complementary VM arises (CVM). It is not necessary for all controllers to have fast performance, many loops have to play the role of accommodating or buffering disturbances. Based on this assumption, we define the Complementary Variability Matrix (*CVM*). The values are computed with actual loop variance ($var_{act,i}$) and the variance of the loop with the worst performance acceptable ($var_{wor,i,j}$). The structure is the same as shown before, and the elements are computed as follows:

$$CVM(i,j) = \frac{\text{var}_{act,i} - \text{var}_{wor,i,j}}{\text{var}_{act,i}} \quad (3)$$

The same procedure as considered earlier can be used to evaluate the Complementary Variability Matrix (*CVM*). Only step 4.1 is replaced by the slowest accepted performance (see Smith, 2002) and the worst accepted performance (var_{wor}) should be quantified.

The proposed computational steps may not be easily applicable in an industrial setting, because the required information (controller and process model) is generally unavailable. The algorithm to compute VM where the controller and plant model are not available is shown in section 3.

2.4 VM Dependence of the System Parameters

From a preliminary inspection, VM seems to be analogous to static the RGA (Skogestad and Postlethwaite, 2005), where only the process static gains have impact in the analysis. However, VM is not only a function of process gains, but also depends on process behavior (dynamics and time delays), disturbance patterns and correlation among the disturbances, controller structure (e.g. PI, PID, MPC, among others), closed loop performance, and best performance achievable. The VM values are specific for each process: even two systems where the models and controllers are the same can have a completely different VM, because of the disturbance pattern.

2.5 Some Peculiar Behaviour

Intuitively, the diagonal elements of the VM should have a positive sign and the off-diagonal elements negative sign, i.e.: improving the performance of a given controller will reduce its variability; and transfer variability to the other loops, increasing their variability. However, this may not always be the case:

Proposition 1*: Diagonal elements of VM can have negative sign, i.e. the performance improvement of a given controller can increase its variability.*

Proof: Consider a SISO system with linear PI type controller that is affected by an output disturbance (*d*). Suppose that the disturbance is a pure white-noise random signal. Considering that *d* is random, it is not possible to predict its future values based on the past values. In this case increasing loop gains will likely increase *y* variability. In this case, the diagonal VM element will have a negative sign.

Proposition 2*: Off-diagonal elements can have positive sign, i.e. the performance increase of a given controller can also decrease the variability in other interacting loops. Typically this happens when interactions help in accommodating disturbances.*

Proof: Consider the triangular system shown in Fig. 3.

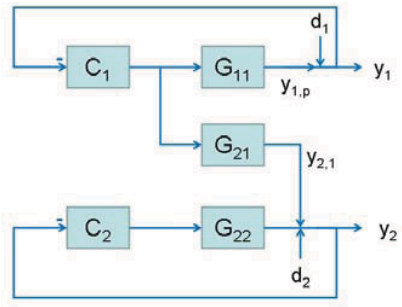


Fig. 3: Schematic representation of the triangular system

Consider the case when C_1 reduces the output variability when it is compared with the open loop case (i.e. $\sigma^2(d_1 + y_{1,p}) < \sigma^2(d_1)$), and upon improving C_1 performance, y_1 will also decrease its variability.

$$\sigma^2(d_1) > \sigma^2(d_1 + y_{1,p1}) > \sigma^2(d_1 + y_{1,p2}) \quad (4)$$

Where p_1 and p_2 are the controllers performance and $p_2 > p_1$ (i.e. closed loop performance in the second scenario (p_2) is faster than p_1). Considering the case when:

$$\begin{aligned} G_{11} &= G_{21} \\ d_1 &= d_2 \end{aligned} \quad (5)$$

Then $y_{1,p} = y_{2,1}$. From the loop 2 and $y_{2,1}$, it is clear that improving the performance of loop 1, will also have the effect of reducing the variability of y_2. This will occur as $y_{2,1}$ will help offset the effect of d_2 (in the same way as $y_{1,p}$ offsets d_1). Thus leads to:

$$\sigma^2(d_2) > \sigma^2(d_2 + y_{2,1,p1}) > \sigma^2(d_2 + y_{2,1,p2}). \quad (6)$$

3. PRACTICAL ISSUES IN COMPUTING VM

3.1 Computing the VM

This section presents the methodology to evaluate VM in industrial settings where process and/or controller models may not be available.

The first analyzed scenario is where a Model Predictive Controller is implemented. In this case, the controller model is not available, because most industrial MPCs are "closed box solutions". However, the plant model is available. In this case, setpoint variations in MPC controllers are quite common, because of the optimization layer. In this scenario, the controller model can be extracted (identified) using the *Asymptotic Method* (Zhu, 1998) or *Subspace Identification* (Overschee and Moor, 1996).

A second scenario contemplates the case where only low order controllers (PI and PID) are present and setpoint activity is available in all loops. For this case, the following steps are contemplated: (I) identify the controller order and parameters (*C*) using *structured target factor analysis* (STFA) (Fotopoulos et al., 1994); (II) estimate the time delay (Tuch et al., 1994); (III) identify the process model (*G*) using *Subspace Identification* (Overschee and Moor, 1996); (IV) identify the disturbance model (*d*) using *Subspace Identification*; (V) with *G*, *C*, and *d* available, the VM can be estimated applying the methodology shown in section 2.3.

Based on our limited experience, we can affirm that the VM is not extremely dependent on the accuracy of the plant and controller. Even for a visible mismatch in the plant model, the obtained results are fairly good, comparing with the case where accurate controller and plant models are available.

3.2 Best and Worst Controller Performances

A natural question that arises is: how can the best and worst performance be computed for a given system? The answer clearly depends on the controller that is implemented on the process.

For MPC controllers, the best achievable performance can be computed using the methodology proposed by Trierweiler and Farina (2003). If the desired performance is attainable, this methodology provides the tuning parameters for the chosen performance. Otherwise, if it's not achievable, the best achievable performance is quantified. In this work, we assume that the "best performance" is based on the open and closed loop rise time ratio, and a convenient value for this ratio is 3.

For low order (PI and PID) decentralized controllers, the best performance can be estimated using the methodology proposed by Faccin and Trierweiler (2004). The worst performance can be evaluated based on the methodology to tune buffer tank controllers (Smith, 2002).

4. QUANTIFYING THE ECONOMIC BENEFITS BASED ON VM

The economic benefits of improving control performance of each loop can be computed in two ways. The first method considers that the best performance can be achieved. In this case the VM can be used as follows. We represent the column j of the VM as VM_j. The economic benefit can be easily quantified using the relationship:

$$CLEB = D \cdot VM \quad (7)$$

where *CLEB* is the *Control Loop Economic Benefit* array. It has the same number of elements as the number of loops in the plant (l).

$$CLEB = \begin{bmatrix} D \cdot VM_1 & D \cdot VM_2 & \cdots & D \cdot VM_l \end{bmatrix} \quad (8)$$

Where D is the array that translates variability reduction into \$ per unit time.

$$D = \begin{bmatrix} D_1 & D_2 & \cdots & D_m \end{bmatrix} \quad (9)$$

where m is the number of main loops in the plant. This array can be quantified as a function of plant throughput increase, utilities reduction, etc. This value can be provided by the commercial department of the plant or the optimization layer weights used in MPC design.

However, as previously mentioned, not all controllers need to have high or tight tuning and the economic benefit, considering the worst performance of each one, can also be quantified. This vector is defined as *Complementary Control Loop Economic Benefit:*

$$CCLEB = \begin{bmatrix} D \cdot CVM_1 & D \cdot CVM_2 & \cdots & D \cdot CVM_l \end{bmatrix} \quad (10)$$

For example, suppose a plant where the VM and D are:

$$VM = \begin{bmatrix} 0.7 & -0.6 \\ -0.3 & 0.8 \end{bmatrix} \quad (11)$$

$$D = [100 \quad 50] \quad (12)$$

the *CLEB* is then be computed as:

$$CLEB = \begin{bmatrix} 55 & -20 \end{bmatrix} \quad (13)$$

The CLEB indicates that improvement in loop 1 performance means increase the plant profitability. However, the opposite behavior is expected when loop 2 performance is improved.

5. CASE STUDIES

5.1 Wood and Berry Distillation Column Model

The pilot-scale distillation column proposed by Wood and Berry (1973) is the first case study. The plant model is given by:

$$\begin{bmatrix} x_D(s) \\ x_B(s) \end{bmatrix} = \begin{bmatrix} \frac{12.8}{16.7s+1}e^{-1s} & \frac{-18.9}{21s+1}e^{-3s} \\ \frac{6.6}{10.9s+1}e^{-7s} & \frac{-19.4}{14.4s+1}e^{-3s} \end{bmatrix} \begin{bmatrix} R(s) \\ S(s) \end{bmatrix} \quad (14)$$

where x_D and x_B are the overhead and bottom products composition, and R and S are the reflux and steam flow rates, respectively. The time constants and time delays are expressed in minutes.

Two decentralized PI type controllers were applied in this case study. The disturbance was generated by passing a random signal through a first order transfer function with unitary gain and 50 minute time constant. The VM analysis of this case study is presented next under 3 scenarios: 1) controller and plant models are assumed to be available; 2) only plant model is available; 3) neither the plant model nor controller models are available. However setpoint activity is assumed. This serves as good excitation for closed loop identification. For case 3), details of closed-loop based subspace identification method are not included here due to lack of space. The PI controllers were tuned to have a performance where the desired closed loop rise time is twice faster the open loop case. We consider here the best achievable performance when the rise time is 6 times faster than open loop.

The *D* vector for this case is hypothetically set as:

$$D = \begin{bmatrix} 100 & 30 \end{bmatrix} \tag{15}$$

In the first scenario, the controller and plant model were available. The VM was computed using the methodology shown in section 2.3.

$$VM = \begin{bmatrix} 0.57 & -0.17 \\ -0.18 & 0.41 \end{bmatrix} \tag{16}$$

The *CLEB* for this case is:

$$CLEB = \begin{bmatrix} 52 & -5 \end{bmatrix} \tag{17}$$

Based on *CLEB*, loop 1 should have its performance improved (top composition), increasing the plant profitability. Loop 2 shows the opposite behavior, improvement in its performance is likely to result in decreased plant profitability.

In the second scenario, the controller model is assumed to be unavailable. Initially, using a scenario where two setpoint variations in each variable are available, the controller model was identified (see section 3.1). In this scenario, the VM was estimated to be:

$$VM = \begin{bmatrix} 0.60 & -0.19 \\ -0.19 & 0.46 \end{bmatrix} \tag{18}$$

Notice that the estimated VM closely matches the true VM shown in (16). In the third scenario, both controller and plant model were identified using closed loop data. The estimated VM for this scenario is:

$$VM = \begin{bmatrix} 0.60 & -0.19 \\ -0.18 & 0.46 \end{bmatrix} \tag{19}$$

Even for this case, where controller and plant model were first identified using subspace identification, a good estimate of VM was obtained.

5.2 Shell Benchmark Process

The Shell Control Problem benchmark was proposed by Prett and Morari (1987). The system is characterized by the high interaction among channels and large time delays.

It involves one heavy oil fractionator. It has three product draws, three side circulating loops and a gaseous feed stream. The system consists of seven measured outputs, three manipulated inputs and two unmeasured disturbances. In this case study, we will reduce the problem to a 3 input and 3 output case. The three controlled variables are: top end point (*y1*); side endpoint (*y2*); bottom reflux temperature (*y3*). The manipulated variables are: top draw (*u1*); side draw (*u2*); bottom reflux duty (*u3*). The system has also two disturbances: upper reflux (*d1*); intermediate reflux (*d2*). The process output can be written as:

$$y = Gu + G_d d \tag{20}$$

Where G is the plant model

$$G = \begin{bmatrix} \frac{4.05}{50s+1}e^{-27s} & \frac{1.77}{60s+1}e^{-28s} & \frac{5.88}{50s+1}e^{-27s} \\ \frac{5.39}{50s+1}e^{-18s} & \frac{5.72}{60s+1}e^{-14s} & \frac{6.9}{50s+1}e^{-15s} \\ \frac{4.38}{33s+1}e^{-20s} & \frac{4.42}{44s+1}e^{-22s} & \frac{7.2}{19s+1} \end{bmatrix} \tag{21}$$

and G_d is the disturbance model:

$$Gd = \begin{bmatrix} \frac{1.2}{45s+1}e^{-27s} & \frac{1.44}{40s+1}e^{-27s} \\ \frac{1.52}{25s+1}e^{-15s} & \frac{1.83}{20s+1}e^{-15s} \\ \frac{1.14}{27s+1} & \frac{1.26}{32s+1} \end{bmatrix} \tag{22}$$

Where the time constant and time delays are reported in minutes. The MPC from Matlab® (MPC toolbox V. 2.2.2) was applied in this study. The analysis for this case is reported under two scenarios: (I) where controller and plant models are available and (II) when both are unavailable.

The *D* vector for this case is (hypothetically set):

$$D = \begin{bmatrix} 100 & 50 & 20 \end{bmatrix} \tag{23}$$

The actual performance in this case was computed based on closed loop rise time when it is set equal to the open loop case. The desired performance for each channel is three times faster than open loop. The VM for this scenario is:

$$VM = \begin{bmatrix} -0.39 & 0.10 & 0.26 \\ -0.12 & -0.16 & 0.28 \\ -0.24 & -0.60 & 0.32 \end{bmatrix} \tag{24}$$

The VM shows that improving the performance of controller 1 and 2 will result in an increase in the variance of all main loops. Retuning loop 1 means increase its variance by 39% and increase in the variances of loops 2 and 3 by 12% and 24% respectively. On the other hand, improvement of loop 3 performance means decrease in its variance by 32% and

corresponding reductions in variances in loop 1 and loop 2 by 26% and 28%, respectively. The *CLEB* for this case is:

$$CLEB = \begin{bmatrix} -50 & -10 & 46 \end{bmatrix} \quad (25)$$

The answer to improving plant profitability lies not only in VM but also the CVM, i.e. some controllers should not have their performance improved, but rather detuned. The CVM for this case is:

$$CVM = \begin{bmatrix} 0.15 & 0.13 & -0.12 \\ 0 & 0.28 & -0.13 \\ -0.30 & -0.29 & -0.44 \end{bmatrix} \quad (26)$$

The *CCLEB* for this case is:

$$CCLEB = \begin{bmatrix} 9 & 21 & -27 \end{bmatrix} \quad (27)$$

The maintenance list for this hypothetical case indicates that the most important controller to maintain or improve performance is loop 3. The second loop in the maintenance list should be loop 2 followed by loop 1.

In the second scenario, both controller and plant are assumed to be unavailable, only setpoint activity is assumed. In this case, the VM is estimated using the procedure shown in section 3.1. The estimated VM is:

$$VM = \begin{bmatrix} -0.42 & 0.06 & 0.26 \\ -0.15 & -0.27 & 0.28 \\ -0.32 & -0.74 & 0.32 \end{bmatrix} \quad (28)$$

Both plant model and controller are identified using subspace identification from Matlab® (system identification toolbox version 6.1.1, function n4sid). Both models have 9 states.

Eq. 28 shows that the estimated VM compared with the original (eq. 24) is fairly good. We attribute the success of this fairly accurate VM estimation to the direct closed loop subspace identification method under reasonable level of setpoint activity.

6. CONCLUDING REMARKS

The main conclusions of the proposed work can be summarized as:

- industrial plants have many loops with considerable potential for performance improvement and therefore a methodology to prioritize loop maintenance is required;
- the concept of Variability Matrix was introduced in this work and has been shown to highlight the potential improvement in each loop and its impact on the whole plant;
- the methodologies to compute VM where neither the controller nor plant model are available has also been presented; in this scenario Subspace Identification can be used; even for this case the methodology has been shown to yield very good results based on closed loop identification;
- the proposed methodology was applied to two case studies providing good results;
- the proposed scenarios where the VM can be computed allows the application of these ideas in an industrial setting.

ACKNOWLEDGMENT

The first two authors wish to thank CAPES, PETROBRAS and FINEP for supporting this work.

REFERENCES

BIALKOWSKI, W. L. (1993) Dreams versus reality: A view from both sides of the gap. *Pulp and Paper Canada,* 94**,** 19-27.

CRAIG, I. K. & HENNING, R. G. D. (2000) Evaluation of advanced industrial control projects: a framework for determining economic benefits. *Control Engineering Practice* 8**,** 12.

FACCIN, F. & TRIERWEILER, J. O. (2004) A Novel Tool for multi-model PID Controller Design. *7th IFAC Symposium on Dynamics and Control of Process Systems.* Boston- USA, IFAC.

FOTOPOULOS, J., GEORGAKIS, C. & STENGER, H. G., JR (1994) Structured target factor analysis for the stoichiometric modeling of batch reactors. *American Control Conference, 1994.*

HUANG, B. & SHAH, S. L. (1999) *Performance assessment of control loops : theory and applications,* London, Springer.

JELALI, M. (2006) An overview of control performance assessment technology and industrial applications. *Control Engineering Practice,* 14**,** 441-466.

MASCIO, R. D. & BARTON, G. W. (2001) The economic assessment of process control quality using a Taguchi-based method. *Journal of Process Control,* 11**,** 8.

MUSKE, K. R. (2003) Estimating the economic benefit from Improved Process Control. Ind. Eng. Chem. Res.,42, 4535-4544.

OVERSCHEE, P. V. & MOOR, B. L. R. D. (1996) *Subspace identification for linear systems : theory, implementation, applications,* Boston, Kluwer Academic Publishers.

PRETT, D. M. & MORARI, M. (1987) *The Shell Process Control Workshop,* Boston, Butterworths.

SKOGESTAD, S. & POSTLETHWAITE, I. (2005) *Multivariable feedback control : analysis and design,* Chichester, England ; Hoboken, NJ, John Wiley.

SMITH, C. A. (2002) *Automated continuous process control,* New York, J. Wiley.

TRIERWEILER, J. O. & FARINA, L. A. (2003) RPN tuning strategy for model predictive control. *Journal of Process Control,* 13**,** 591-598.

TUCH, J., FEUER, A. & PALMOR, Z. J. (1994) Time delay estimation in continuous linear time-invariant systems. *IEEE Transactions on Automatic Control,* 39**,** 823-827.

WOOD, R. K. & BERRY, M. W. (1973) Terminal composition control of a binary distillation column. *Chemical Engineering Science,* 28**,** 11.

ZHU, Y. (1998) Multivariable process identification for MPC: the asymptotic method and its applications. *Journal of Process Control,* 8**,** 15.

Soft sensor models: Bias updating revisited

André D. Quelhas* and José Carlos Pinto**

**Petrobras – Corporate University, Rio de Janeiro - Brazil*
(Tel: +55-21-3487-3467; e-mail:quelhas@ petrobras.com.br).
***Chemical Engineering Program, COPPE-UFRJ, Rio de Janeiro - Brazil*
(e-mail:pinto@peq.coppe.ufrj.br)

Abstract: Bias updating is a widespread adaptive procedure to allow inference models to pursue time variant features of a real world process. The aim of this work is to clarify the statistical consequences of bias updating to soft sensor estimates as well to point up the need of careful analysis of the effect of unmeasured disturbances on the true values of the variable of interest. It is shown that bias updated inferences are unbiased estimates of the true value but yields estimates whose variance are 100% larger than the ones obtained with no use of bias updating. It is suggested the use of a weighting factor to bias updating in order to balance statistical benefits and penalties. A case study of a soft sensor for weathering of LPG in oil refinery exemplifies the concepts discussed.

Keywords: Soft sensor, Bias Updating, Error Analysis, Statistics.

1. INTRODUCTION

The main goal of an industry is to operate as close as possible to the point where profit is maximum. It means that there should be no off-spec product and the lowest degree of product quality give-away should be achieved. Maximum profit is also related to the fact that the set of manipulated variables leads to lower costs by minimizing use of heat, steam, electricity, water etc..

It may be hard if not impossible to accomplish this goal. Real processes are likely to be nonlinear and highly integrated causing modeling and identification prone to errors. In addition, long term operation makes processes more susceptible to hardware upsets (corrosion, fouling, mechanical failures) and to experience environmental disturbances as well qualitative/quantitative changes in physical-chemical properties of feed streams.

Accurate knowledge of process actual model structure and parameters is essential if one intends to predict future states (for control and optimizing) or to diagnose safety risks. Unfortunately many relevant process variables are not available as frequently as desirable or even not available at all. For example, it is very common that physical-chemical properties related to quality control are measured by laboratory tests performed with a very low frequency when compared to process variables acquired by online sensors. Such process with differing sample rates for measured variables are known as multirate process (Ragahavan *et al.*, 2006).

Most of times the long period of time to be awaited before new information about low frequency variables become available is unacceptable. It is necessary to make use of some inferential knowledge based on high frequency information about the process. If a sufficiently accurate model is available, the variable of interest can be estimated from high frequency process measurements $\mathbf{x}$ as long as model structure and parameters $\boldsymbol{\alpha}$ are known:

$$\hat{y} = f(\mathbf{x}, \boldsymbol{\alpha}) + bias \tag{1}$$

Every time a new measurement of the *true* value of y is available, an adaptive procedure can be used to adapt the inferential model. The only parameter updated through this one parameter correction is the independent coefficient in (1): $bias = y - f(\mathbf{x}, \boldsymbol{\alpha})$. This simple strategy is very common in industry as well in literature for optimizing purposes (Mercangöz and Doyle 2008; Jesus 2004; Singh 1997) or for soft sensors inferences (Sharmin *et al.* 2006; Mu *et al.* 2006; Tran *et al*. 2005).

Some questions should be posed regarding the use of inferences as (1) for anyone who has to cope with a multirate process:

- What is the best model structure f($\mathbf{x}$,$\boldsymbol{\alpha}$)?
- How often should bias be updated?
- How are inference errors affected by bias updating?
- What are the effects of unmeasured disturbances on inference errors?

Those questions usually receive unequal importance. A lot of effort has been spent along time to answer the first question. Models have progressively become more complex by using the mathematical weaponry of process modeling

(multivariable regression, PCA, neural networks, fuzzy logic). The second question is often answered based on practical matters as availability of laboratory technicians. The last two questions are normally disregarded in spite of their huge consequences on the estimates.

The aim of this work is to pay attention to those usually forgotten questions by remembering the mathematical considerations implicit in models as (1) and answering, from a statistical point of view, what the benefits and penalties of bias updating are.

2. MATHEMATICAL FOUNDATIONS OF BIAS UPDATING

For a steady state system, the generic mathematical relationship linking the output variable, y, and all pertinent process variables, **w**, required by fundamental physical laws, may be expressed as:

$$F(y, \mathbf{w}, \mathbf{c}) = 0 \quad (2)$$

where **w** represents the NW necessary variables to perfectly predict the unknown behavior of y given the NC constants in the vector of parameters, **c**.

Two practical reasons explain why it is unlike that any real model would incorporate the whole set of NW necessary variables. The first one is the fact that NW may be large and would conflict with science´s parsimony principle. In this sense, a less complete description would be acceptable in a trade-off for simplicity under a certain allowable tolerance. The other reason is that several of the NW variables either are not measured or are not considered relevant by the scientist due to a methodological error.

Taking these reasons under consideration one can split **w** into the subsets **x** and **z**. The first subset contains the NX measured variables that were chosen as relevant for the model. The second subset contains the remaining NZ = NW-NX variables. It contains measured and unmeasured variables that should be part of a perfect model but were set apart. The complete description of the system behavior is then expressed as:

$$F(y, \mathbf{x}, \mathbf{z}, \mathbf{c}) = 0 \quad (3)$$

In the process of justifying the possibility of a correction as proposed in (1) it is required that (3) be partially separable with respect to addition at least with respect to y. It requires that $(1/F)\partial \exp(F)/\partial y$ depends only on y (Viazminsky 2008). If this condition is satisfied one can express (3) as:

$$g(y) = F_1(\mathbf{x}, \mathbf{z}, \mathbf{c}_1) \quad (4)$$

Additionally, if the inverse function g^{-1} exists, then:

$$y = g^{-1}(F_1(\mathbf{x}, \mathbf{z}, \mathbf{c}_1)) = F_2(\mathbf{x}, \mathbf{z}, \mathbf{c}_2) \quad (5)$$

Physical knowledge or empirical insight may lead to an attempt to predict *y* based on measurements **x** and parameters $\boldsymbol{\alpha}$ by means of a model $f(\mathbf{x}, \boldsymbol{\alpha})$. If **z** is an empty set and the whole influence of **x** on *y* is taken into account by $f(\mathbf{x}, \boldsymbol{\alpha})$ we have a perfect model. Otherwise one should expect a relationship as (6), where $F_3(\mathbf{x}, \mathbf{z}, \mathbf{c}_3)$ plays the role of bias as in (1). It should be noticed that (6) is derived from (5) if $F_3(\mathbf{x}, \mathbf{z}, \mathbf{c}_3)$ is a separable function with respect to the set **z**.

$$y = f(\mathbf{x}, \boldsymbol{\alpha}) + F_3(\mathbf{x}, \mathbf{z}, \mathbf{c}_3) \quad (6)$$

The model built by the experimenter is $f(\mathbf{x}, \boldsymbol{\alpha})$. The invisible part of the true model is $F_3(\mathbf{x}, \mathbf{z}, \mathbf{c}_3)$. This term is captured by the bias term in a very common pragmatic approach assuming the form (1).

Inference structure (6) is very attractive but it is valid only if the assumptions that allowed disregarding more generalized expressions (3)-(5) are true. If not, there will be no guarantee that successive inferred values will express the true values y even if no further disturbances alter the values of the set **z**. This can be seen by comparing two simple models. One represents a model as expressed in (5) (type A model) and the other one represents the less generic model expressed in (6) (type B model), for instance:

type A true model: $y = (x+z)/x$

type B true model: $y = x + z$

It should be noticed that the type B true model in this example shows no dependence of F_3 on x. This class of true models yields the best possible performance for an adaptive experimental model as (1).

Assuming that: 1) experiments to identify the inference $f(x, \alpha)$ were carried out under controlled conditions in order to keep z at a constant value z_0 in both cases and 2) perfect model identification led to inferences with the same mathematical structure than true models:

type A inferred model: $\tilde{y} = (x+ z_0)/x$

type B inferred model: $\tilde{y} = x + z_0$

If the inferred models were parameterized by means of proper statistical criticism both inferred models will adequately represent the behavior of the variable of interest. However, as time passes, it is possible that z assumes values different of the one kept controlled along identification phase. So, if z assumes the value z_1 and $x=x_1$ at the moment of correction in both cases, according to the bias updating routine:

type A true value: $y_1 = (x_1+z_1)/ x_1$,

type A inferred value: $\tilde{y}_1 = (x_1 + z_0)/ x_1$

$\Rightarrow$ bias = $y_1 - \tilde{y}_1 = (z_1 - z_0)/x_1$

corrected inference: $\hat{y}_1 = (x + z_0)/x + ((z_1 - z_0)/x_1)$

type B true value: $y_1 = x_1 + z_1$,

type B inferred value: $\tilde{y}_1 = x_1 + z_0$

$\Rightarrow$ bias = $y_1 - \tilde{y}_1 = z_1 - z_0$, corrected inference: $\hat{y}_1 = x + (z_1 - z_0)$

It is clear that, after bias correction, inferences derived from type B models will produce results as close to the truth as they were before the change of z value as long as this variable is kept constant from this change on. On the other hand, inferences derived from type A models will not behave this way because accuracy of the corrected inference will be affected not only by further changes of z value but also by additional changes in the x value because the nonlinear behavior is not captured by a single point correction.

3. BIAS UPDATING PROCEDURE

In order to describe the behavior of predictions of the value *y* along time it is interesting to write inference model to allow time course to be taken into account:

$$\tilde{\mathbf{y}} = f(\mathbf{X}, \boldsymbol{\alpha}), \quad \mathbf{X} = \begin{bmatrix} \mathbf{x_1} \\ \mathbf{x_2} \\ \dots \\ \mathbf{x_{NS}} \end{bmatrix}, \quad \mathbf{x_i} = \begin{bmatrix} x_{i1} & x_{i2} & x_{i3} & \dots & x_{i,NX} \end{bmatrix},$$

$$\tilde{\mathbf{y}} = \begin{bmatrix} \tilde{y}_1 \\ \tilde{y}_2 \\ \dots \\ \tilde{y}_{NS} \end{bmatrix} \tag{7}$$

where NS is the number of time samples of the process variable signals.

Corrected values of *y* along time are obtained from bias updating according to:

$$\hat{\mathbf{y}} = \tilde{\mathbf{y}} + \mathbf{bias} \tag{8}$$

where the array of time values of bias is built according to:

$bias_1 = 0$

$bias_k = \left(y_k^m - \tilde{y}_k\right) s_k + (1 - s_k) bias_{k-1}$

leading to:

$$\mathbf{bias} = \begin{bmatrix} 0 \\ \left(y_2^m - y_2\right) s_2 + (1 - s_2) bias_1 \\ \left(y_3^m - y_3\right) s_3 + (1 - s_3) bias_2 \\ \dots \\ \left(y_{NS}^m - y_{NS}\right) s_{NS} + (1 - s_{NS}) bias_{NS-1} \end{bmatrix} \tag{9}$$

Vector $\mathbf{y^m}$ contains measurements of the true values $\mathbf{y}$ sampled with period at a lower rate, T_{meas}, than primary variables of the model. Vector $\mathbf{s}$ is a binary set that indicates when true values $\hat{y}$ are available:

$$\mathbf{y^m} = \begin{bmatrix} [0]_{1x(Tmeas-1)} & y_{Tmeas} & [0]_{1x(Tmeas-1)} & y_{2Tmeas} & \dots \end{bmatrix}^T$$

$$\mathbf{s} = \begin{bmatrix} [0]_{1x(Tmeas-1)} & 1 & [0]_{1x(Tmeas-1)} & 1 & \dots \end{bmatrix}^T$$

4. STATISTICAL IMPACT OF BIAS UPDATING

Although equations (7-9) indicate the *modus operandi* of inference correction, it is not clear how our expectation about error values is affected. Since corrections are made at a low frequency the duration of their benefits will be affected by the probability of occurrence of new disturbances before a new gold standard measurement is ready, thus making possible another correction. A reasonable question would be: what benefits are obtained with periodic bias updating comparing with no bias correction at all?

In fact, bias updating and no updating schemes are extreme points of a continuous range of possible single point corrections. Considering the weight parameter $\varphi \in \Re$, $\varphi \subset [0\ 1]$, the time values $\bar{\mathbf{y}}$ are a weighted mean of bias corrected values (8) and values from the original inference model (7):

$$\bar{\mathbf{y}} = \varphi \hat{\mathbf{y}} + (1 - \varphi) \tilde{\mathbf{y}} \tag{10}$$

If samples of true values are taken with period T_{meas} the n^{th} element suffers the effects of the last bias updating made at sample $i = \text{int}(n/T_{meas})T_{meas}$, where int(x) retains the integer part of the floating point real number x:

$$\overline{y_n} = \varphi\left(\tilde{y}_n + bias_n\right) + (1 - \varphi)\tilde{y}_n \tag{11}$$

$$\overline{y_n} = \varphi\left(\tilde{y}_n + y_i - \tilde{y}_i\right) + (1 - \varphi)\tilde{y}_n = \tilde{y}_n + \varphi\left(y_i - \tilde{y}_i\right) \tag{12}$$

Inference error at the n^{th} element will be:

$$\varepsilon_n = y_n - \overline{y_n} = y_n - \widetilde{y}_n - \varphi y_i + \varphi \widetilde{y}_i \quad (13)$$

Since n^{th} and i^{th} elements of the true values come from the same sample space as well n^{th} and i^{th} elements of the inferred values, their statistical moments are the same, i.e., $E[y_n] = E[y_i]$ and $E[\widetilde{y}_n] = E[\widetilde{y}_i]$. Dropping indexes to simplify notation, it is possible to say that the expected error value is:

$$E[\varepsilon] = (1-\varphi)E[y] + (\varphi - 1)E[\widetilde{y}] \quad (14)$$

$$if \begin{cases} \varphi = 1, & E[\varepsilon] = \varepsilon_{\min} = 0 \\ \varphi = 0, & E[\varepsilon] = \varepsilon_{\max} = E[y] - E[\widetilde{y}] \end{cases}$$

It can be seen that the bias update scheme expressed in (8) ($\varphi = 1$) guarantees mean error value of zero if length of **y** tends to infinity. If no correction is made ($\varphi = 0$), long term error mean depends on the ability of model $f(\mathbf{X}, \boldsymbol{\alpha})$ to be an unbiased estimate of the true value. It is also possible to investigate the dependence of error variance on the choice of φ. From (13) it is possible to write:

$$\mathrm{var}(\varepsilon_n) = \mathrm{var}(y_n - \widetilde{y}_n - \varphi y_i + \varphi \widetilde{y}_i) \quad (15)$$

$$\begin{aligned} \mathrm{var}(\varepsilon_n) = {} & \mathrm{var}(y_n) + \mathrm{var}(\widetilde{y}_n) + \varphi^2 \mathrm{var}(y_i) + \varphi^2 \mathrm{var}(\widetilde{y}_i) \\ & - 2\,\mathrm{cov}(y_n, \widetilde{y}_n) + 2\varphi\,\mathrm{cov}(y_n, \widetilde{y}_i) - 2\varphi\,\mathrm{cov}(y_i, \widetilde{y}_n) \\ & - 2\varphi^2\,\mathrm{cov}(y_i, \widetilde{y}_i) \end{aligned} \quad (16)$$

For the same reason explained above $\mathrm{var}(y_n) = \mathrm{var}(y_i)$ and $\mathrm{var}(\widetilde{y}_n) = \mathrm{var}(\widetilde{y}_i)$, making it more convenient to drop subscripts and simplify (16):

$$\begin{aligned} \mathrm{var}(\varepsilon) = {} & \mathrm{var}(y) + \mathrm{var}(\widetilde{y}) + \varphi^2(\mathrm{var}(y) + \mathrm{var}(\widetilde{y})) - 2\,\mathrm{cov}(y, \widetilde{y}) \\ & - 2\varphi^2\,\mathrm{cov}(y, \widetilde{y}) \end{aligned} \quad (17)$$

At the extreme points of φ:

$$if \begin{cases} \varphi = 1, & \mathrm{var}(\varepsilon) = v\varepsilon_{\max} = 2\,\mathrm{var}(y) + 2\,\mathrm{var}(\widetilde{y}) - 4\,\mathrm{cov}(y, \widetilde{y}) \\ \varphi = 0, & \mathrm{var}(\varepsilon) = v\varepsilon_{\min} = \mathrm{var}(y) + \mathrm{var}(\widetilde{y}) - 2\,\mathrm{cov}(y, \widetilde{y}) \end{cases}$$

With respect to the error variance the progressive updating ($\varphi = 1$) doubles the value obtained when no correction is made ($\varphi = 0$). Confronting this result with the expected value of the error one can see that bias updating is associated with an expectation of unbiased mean value of estimates but it also causes a 100% increase in error variance. There would be a choice of φ to cope with these consequences? In order to answer this question it is necessary to create a single objective function that combines both effects.

As an example, a possible choice for such function could be $\psi = E[\varepsilon] + \mathrm{var}(\varepsilon)$, choosing φ that minimizes its value. However this function is too dependent of the problem specificities and units of measurement. In fact even the choice the objective function depends on the problem to be solved and on the needs of the plant personnel in order to fulfill several goals related to the industrial process.

Taking this into consideration, it is suggested a very simple objective function, derived from the previous one. It represents an attempt to equalize the importance of the effects of φ regarding each statistical moment. Such function could assume the normalized form:

$$\psi = E[\varepsilon]_{norm} + \mathrm{var}(\varepsilon)_{norm} \quad (18)$$

where

$$E[\varepsilon]_{norm} = \frac{E[\varepsilon] - \varepsilon_{\min}}{\varepsilon_{\max} - \varepsilon_{\min}} \quad (19)$$

and

$$\mathrm{var}(\varepsilon)_{norm} = \frac{\mathrm{var}(\varepsilon) - v\varepsilon_{\min}}{v\varepsilon_{\max} - v\varepsilon_{\min}} \quad (20)$$

Substituting (19-20) in (18):

$$\begin{aligned} \psi = {} & \frac{(1-\varphi)E[y] + (\varphi - 1)E[\widetilde{y}]}{E[\hat{y}] - E[y]} \\ & + \frac{\varphi^2(\mathrm{var}(y) + \mathrm{var}(\widetilde{y})) - 2\varphi^2\,\mathrm{cov}(y, \widetilde{y})}{\mathrm{var}(y) + \mathrm{var}(\widetilde{y}) - 2\,\mathrm{cov}(y, \widetilde{y})} \end{aligned} \quad (21)$$

The choice of φ is made in order to minimize ψ and is represented by the solution of:

$$\frac{\partial \psi}{\partial \varphi} = 1 - \varphi - \varphi^2 = 0 \;\Rightarrow\; \varphi = 1/2 \quad (22)$$

It is interesting to see how formalism of (21) and (22) conducts to a common sense value of ½ for the weighting factor in this case.

5. CASE STUDY

In this section it will be shown the statistical features of bias updating in a soft sensor to be implemented in an oil refinery. The process unity at study is a FCC debutanizer showed in figure 1. In order to improve quality control of liquefied petroleum gas (LPG) it is desirable to have online information about the relative amount of molecules with more than four carbon atoms present on LPG stream. A laboratory or field test usually carried out a few times a day measures weathering of LPG, expressed in temperature units, which is correlated to the ratio of heavier molecules.

An empirical mathematical model of LPG weathering based on NX = 3 process variables feeds the model predictive control of the process unity with inferred values along time as in (7).

For the purposes of this work, actual behavior of the unity is represented by data from customized process simulation software. The discrete mathematical space of operating scenarios has its basis formed by the N_{inp} = 4 process simulation input parameters as shown in figure 1.

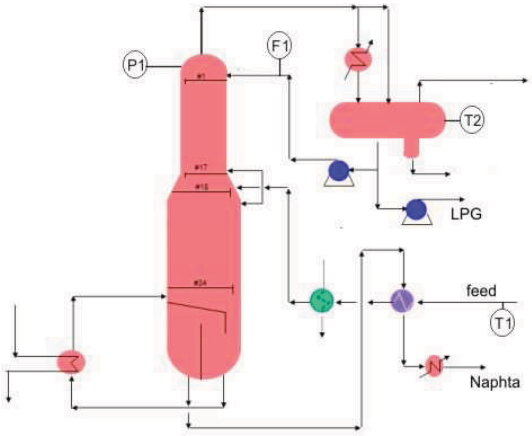


Figure 1 – FCC debutanizer. Process variables used as inputs for the process simulator: P1, F1, T1, T2.

The subregion of operation considered for analysis was the regular mesh **S** (N_{sc} x N_{inp}) of equally spaced points around nominal condition of operation. This region of operation induces the subregion $\boldsymbol{\chi}$ (N_{sc} x NX) of the input variables of the empirical model of weathering. For simulation of long term operation a string of scenarios, $\mathbf{S^{str}}$ (L_{sc} x N_{inp}), representing the time course of conditions of operation, was assembled:

$ind_i \sim \text{Unif}(1, N_{sc})$; $ind_i \in \mathbb{N}$; i = [1 2 ... L_{sc}]

Each choice ind_i is a uniformly distributed random variable that indicates where, in the subregion **S**, is the i^{th} line of $\mathbf{S^{str}}$ and, as consequence, maps **X** (L_{sc} x NX) as in (7):

$$\begin{aligned} &\mathbf{S}(ind_i, \mathbf{j}) \rightarrow \boldsymbol{\chi}(ind_i, \mathbf{k}) \\ &\mathbf{S^{str}} = \mathbf{S}(\mathbf{ind}, \mathbf{j}) \rightarrow \mathbf{X} = \boldsymbol{\chi}(\mathbf{ind}, \mathbf{k}) \\ &\mathbf{j} = [1 \quad 2 \quad \ldots \quad N_{inp}],\ \mathbf{k} = [1 \quad 2 \quad \ldots \quad NX] \end{aligned} \tag{22}$$

Since quality of the feed is a major unmeasured disturbance the set of variables **z** is represented by the ratio of the slope of the true boiling point curve of the actual feed related to the one at nominal operating condition. If feed stream may have three different compositions symmetrically disturbed:

$\mathbf{r}$=[0.95 1 1.05]

$ind_i \sim \text{Unif}(1, 3)$; $ind_i \in \mathbb{N}$; i = [1 2 ... L_{sc}]

$$\mathbf{z} = \mathbf{r}(\mathbf{ind}) \tag{23}$$

In order to allow a better understanding of the different effects observed in the results there will be considered two cases of study. In the more generic case A it is supposed that the set of variables **z** is represented by (23) and that the inference model is the actual one used in industrial practice. Case B will also take disturbances as in (23) into account but it is supposed that the inference model was perfectly modeled in the absence of disturbances. It is perfect in the sense that all the effects of the model input variables perfectly propagate to the output variable. In other words, at $z=z_{nominal}$, $F_3 = F_3(z,c3)$ as in (6) and the inference is correct for any value of **x**.

As it can be seen in figure 2, in both cases bias updating procedure yields an expected mean value of zero although values show less dispersion when no bias correction is used.

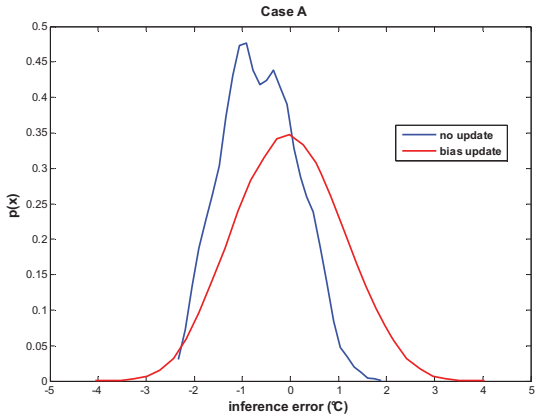


Figure 2 – Estimated probability density function of inferred weathering values for case A.

Estimated probability density function for case B (fig. 3) shows additional features. In this case it is clear that, with no update, the inference will be correct every time $z=z_{nominal}$ whatever the **x** values. The two triangular areas under the blue line around the central peak in figure 3 are originated when $z= z_{nominal} \pm \Delta z$. It should be noticed that the fact that those areas are not as thin as the central peak is due to the dependence of F3 (6) on **x**. When bias update is implemented, two more regions appear as well all regions become flatter. It is because bias expected values will be the result of the difference of all possible two random samples respectively chosen from the sample space of the non corrected inferred values and from the sample space of true values. These bias values will be summed to the inferred ones creating the oscillations of the red line at extreme inference errors observed in figure 3.

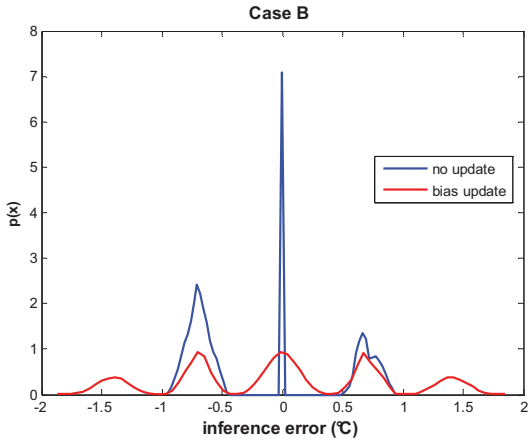


Figure 3 – Estimated probability density function of inferred weathering values for case B.

6. CONCLUSIONS

This work addressed the problem of continuous time monitoring in processes with differing sample rates for measured variables. Bias updating is a common adaptive procedure to periodically correct soft sensor models estimates. It was shown that this strategy is associated with long term zero mean error but at very high cost of 100% increase in variance of estimates. Our intention was to shown that a procedure to implement periodical parameter update should be problem-specific. It means to take into account statistical impact on estimates based on prior knowledge of probability density of disturbances as well error magnitude of soft sensor estimates.

ACKNOWLEDGMENT

The authors are grateful to Eng. Cristina Neves Passos, from Petrobras, for providing simulation files from FCC plant that were used as templates for our work.

REFERENCES

Jesús, R.A.J. (2004), *Modelación y Optimización del Mezclado de Petróleo Crudo con Redes Neuronales*, Master thesis, Centro de Investigación y de Estudos Avanzados del Instituto Politécnico Nacional de Mexico.

Mercangöz, M., Doyle, F.J. (2008), Real-time optimization of the pulp mill benchmark problem, *Computers and Chemical Engineering*, 32, 789-804.

Mu, S., Zeng, Y., Liu, R., Wu, P., Su, H., Chu, J. (2006), Online dual updating with recursive PLS model and its application in predicting crystal size of purified terephthalic acid (PTA) process, *Journal of Process Control*, 16, 557-566

Ragahavan, H.; Tanrigala, A.K.; Gopaluni, B.; Shah, S. (2006), Identification of chemical processes with irregular output sampling, *Control Engineering Practice*, 14, 467-480.

Sharmin, R., Sundararaj, U., Shah, S., Griend, L.V., Sun, Y. (2006), Inferential sensors for estimation of polymer quality parameters - Industrial application of a PLS-based soft sensor for a LDPE plant, *Chemical Engineering Science*, 61, 6372-6384.

Singh, A. (1997). *Modeling and Model Updating in the Real-Time Optimization of Gasoline Blending*, Master thesis, University of Toronto.

Tran, M., Varvarezos, D.K., Nasir, M. (2005), The importance of first-principles, model-based steady-state gain calculations in model predictive control - a refinery case study, *Control Engineering Practice*, 13, 1369-1382.

Viazminsky, C.P. (2008). Necessary and sufficient conditions for a function to be separable. *Applied Mathematics and Computation*, 204, 658-670.

Data derived analysis and inference for an industrial deethanizer

Francesco Corona [*] **Michela Mulas** [**] **Roberto Baratti** [***]
Jose A. Romagnoli [***,1]

[*] *Dept. of Computer and Information Science, Helsinki University of Technology, Finland, (e-mail: francesco.corona@hut.fi).*
[**] *Dept. of Bio and Chemical Technology, Helsinki University of Technology, Finland, (e-mail: michela.mulas@hut.fi)*
[***] *Dept. of Chemical Engineering and Materials, University of Cagliari, Italy, (e-mails: baratti@dicm.unica.it, jose@lsu.edu)*

Abstract: In this paper, we present an application of data derived approaches for analyzing and monitoring an industrial deethanizer column. The discussed methods are used in visualizing process measurements, extracting operational information and designing an estimation model. Emphasis is given to the modeling of the data obtained with standard paradigms like the Self-Organizing Map (SOM) and the Multi-Layer Perceptron (MLP). The SOM and the MLP are classic methods for nonlinear dimensionality reduction and nonlinear function estimation widely adopted in process systems engineering; here, the effectiveness of these data derived techniques is validated on a full-scale application where the goal is to identify significant operational modes and most sensitive process variables before developing an alternative control scheme.

Keywords: Process monitoring, Process supervision, the Self-Organizing Map

1. INTRODUCTION

A modern process plant is under tremendous pressure to maintain and improve product quality and profit under stringent environmental and safety constraints. For efficient operation, any decision-making action related to the plant operation requires the knowledge of the actual state of the process. The availability of easily accessible displays and intuitive knowledge of the states is thus indispensable, with immediate implications for profitability, management planning, environmental responsability and safety.

Due to the advances in measuring and information technology, historical data are available in abundance. Remarkable characteristics of the data acquired in industrial facilities are redundancy and possibly insignificance, not to mention the presence of disturbances that corrupt the measurements. Very often, the amount and quality of the data together with their high-dimensionality can be a limiting factor for the analysis; therefore, it is necessary the availability of effective methods that: i) model the data to extract the structures existing in the measurements, ii) identify and reconstruct the most relevant structures for the scope at hand and, iii) allow for easily interpretable displays where the states' information is presented to the plant operators. Intuitive knowledge of all visited states is invaluable for safe plant operation and trustworthy methods become necessary when considering statistical process monitoring as part of a supervision and control strategy.

In this paper, we discuss the implementation and direct application of a strategy to model, visualize and analyze the information encoded in industrial process data. The approach is based on a classical machine learning method for dimensionality reduction and quantization, the Self-Organizing Map, SOM (Kohonen, 2001). The SOM combines many of the main properties of other general techniques and shares many commonalities with two standard methods for data projection (Principal Components Analysis, PCA (Jolliffe, 2002)) and clustering (K-means, (Hartigan et al., 1979)). In addition, the SOM is also provided with a set of tools that allow for efficient data visualization in high-dimensional settings.

The use of the Self-Organizing Map in the exploratory stage of data analysis is discussed in (Kaski, 1997; Vesanto, 2002) and it is widely employed in many fields. In general terms, the main contributions in applying the SOM on industrial process data are collected by Alhoniemi (2002) and Laine (2003), whereas more domain specific developments can be found in the SOM's bibliography (Oja et al., 2003). Here, the SOM is used as a framework for the identification of the process modes with their time of occurence and present the information on simple displays.

To support the presentation, the analysis is discussed on a full-scale deethanizer where the goal is to identify significant operational modes and most sensitive process variables before developing an alternative control scheme. The study relies on an regression model for estimating an important quality variable (the ethane concentration in the bottom) otherwise difficult to measure in real-time from a set of easily measurable process variables. Inference is based on the Multi-Layer Perceptron Haykin (1998).

[1] On leave from the Department of Chemical Engineering, Louisiana State University, Baton Rouge LA 70803, USA.

2. THE SELF-ORGANIZING MAP

The Self-Organizing Map (Kohonen, 2001) is an adaptive formulation of vector quantization performing in unison:

- a reduction of the data dimensionality by projection; that is, the reduction of the dimensionality of the data by mapping all the observations onto a meaningful subspace with lower dimensionality;
- a reduction of the amount of data by clustering; that is, the retention of the original dimensionality of the data space while reducing the amount of observations by prototyping them by similarity.

The SOM nonlinearly projects vast quantities of high-dimensional data onto a low-dimensional array of few prototypes in a fashion that aims at preserving the topology of the observations. By choosing a conventional bi-dimensional array of prototypes, the main advantage of the map is in a wealth of visualization techniques that allows the analysis of the structures existing in the data.

The following overviews the SOM algorithm and its analogies with other projection and clustering methods. A brief presentation of the most common SOM-based visualization methods for exploratory data analysis is also reported.

Algorithm and properties The basic Self-Organizing Map consists of a low-dimensional and regular array of K nodes, where a prototype vector $\mathbf{m}_k \in \mathbb{R}^p$ is associated with each node k. Each prototype acts as an adaptive model vector for the N observations $\mathbf{v}_i \in \mathbb{R}^p$. During the computation of the SOM, the observations are mapped onto the array of nodes and the model vectors adapted according to:

$$\mathbf{m}_k(t+i) = \mathbf{m}_k(t) + \alpha(t) h_{k,c(\mathbf{v_i})} \Big(\mathbf{v}_i(t) - \mathbf{m}_k(t) \Big). \quad (1)$$

In the learning rule in Equation 1, t denotes the discrete-time coordinate of the mapping steps and $\alpha(t) \in (0,1)$ is the monotonically decreasing learning rate. The scalar multiplier $h_{k,c(\mathbf{v_i})}$ denotes a neighborhood kernel centered at the Best Matching Unit (BMU); that is, at the model vector $\mathbf{m}_c(t)$ that, at time t, best matches with the observation vector $\mathbf{v}_i$. The matching is based on a competitive criterion on the Euclidean metric $d(\mathbf{m}_k(t), \mathbf{v}_i(t))$, for all $k = 1, \ldots, K$. At each step t, the BMU is thus the prototype $\mathbf{m}_k(t)$ that is the closest to observation $\mathbf{v}_i(t)$:

$$c(t) = \underset{k}{\operatorname{argmin}} \Big(d(\mathbf{m}_k(t), \mathbf{v}_i(t))^2 \Big), \ \forall k \text{ and } \forall i. \quad (2)$$

The kernel $h_{k,c(\mathbf{v_i})}$ centered at $\mathbf{m}_c(t)$ is often a Gaussian:

$$h_{k,c(\mathbf{v_i})} = \exp\Big(- \frac{||\mathbf{r}_k - \mathbf{r}_c||^2}{2\sigma^2(t)} \Big), \quad (3)$$

where the vectors $\mathbf{r}_k$ and $\mathbf{r}_c$ represent the geometric location of the nodes on the array and $\sigma(t)$ denotes the monotonically decreasing width of the kernel. The effect of the kernel decreases with the distance from the BMU.

The SOM is computed recursively for each observation. As $\alpha(t) h_{k,c(\mathbf{v_i})}$ tends to zero with t, the set of prototype vectors $\{\mathbf{m}_k\}_{k=1}^K$ are adaptively updated to represent similar observations in $\{\mathbf{v}_i\}_{i=1}^N$, and converge toward their asymptotic limits. The resulting model vectors learn a nonlinear manifold in the original embedding space such that the relevant topological and metric properties of the observations are preserved on the map. Thus, the SOM is to be understood as an ordered image of the original high-dimensional data modeled onto a low-dimensional manifold, where the complex data structures are represented by simple geometric relationships.

A rigorous analysis of the SOM has demonstrated difficult. However, in the case of the basic algorithm with a fixed kernel function, also the SOM algorithm can be understood from the optimization of a cost function:

$$\mathtt{E(SOM)} = \sum_{i=1}^{N} \sum_{k=1}^{K} h_{k,c(\mathbf{v_i})} d(\mathbf{m}_k, \mathbf{v}_i)^2. \quad (4)$$

The cost function in Equation 4 is closely related to the objective optimized with the K-mean algorithm (Lloyd, 1982). The only difference is in the neighborhood function that smoothly weights all the distances between the observations and the prototypes, instead of just the closest one. In that sense, the SOM operates as the conventional clustering method where the width of the kernel is zero. Moreover, there is no need to explicitly specify the number of taxonomies; in fact, the number of prototypes in the SOM can be chosen without any specific concern on the actual number of clusters. The SOM has also neat projection properties. In fact, the cost in Equation 4 closely resembles the objective optimized by Curvilinear Components Analysis CCA (Demartines et al., 1997); CCA is a modification of metric Multi-Dimensional Scaling MDS (Cox et al., 2000) and Principal Components Analysis PCA (Jolliffe, 2002). Similarity is in the decreasing and smoothing nature of the neighborhood function that emphasizes smaller distances in the projection. Conversely, the notion of locality in the SOM does not correspond to the global concern on small distances characterizing CCA.

Data exploration methods In the typical case of projections onto 2D arrays, the SOM offers excellent techniques for data exploration. In that sense, the approach to data analysis with the SOM is mainly visual and focuses on the low-dimensional displays specifically designed for the map.

The data visualization techniques based on the SOM assume that the prototype vectors are representative models for groups of similar observations, and projecting the data onto the low-dimensional array allows for an efficient display of the dominant relationships existing between them. For instance, the displays permit to identify the shape of the data distribution, cluster borders, projection directions and dependencies between variables. The visualizations techniques considered here are i) the component planes and ii) the distance matrix. Such techniques were thoroughly studied by Kaski (1997) and Vesanto (2002).

A component plane shows on the SOM's array the coordinates of the prototype vectors along a specific direction in the data embedding space; that is, each component plane is associated to one original variable and there are as many planes as directions in the embedding. The coordinate values are encoded into gray levels or colors, and the area of each unit on the array is dyed with the color associated to the component value. A component plane thus displays the distribution of the corresponding variable among the prototype vectors. The component planes are useful in order to visually identify possible dependencies between variables. The dependencies between variables can be seen as similar patterns (the colors corresponding to the values

of the variables) in identical locations on the component planes. Such representations can be also used to quantify dependencies. In that sense, the SOM reduces the effect of noise and outliers in the observations and, therefore, may actually make any existing dependence simpler to detect.

A distance matrix visualizes on the SOM's array the average distance between each prototype vector and its adjacent neighbors. In a distance matrix, distances are encoded into gray levels or colors and each unit on the array is dyed with the color associated to the distance with the neighbors. The most widely used distance matrix for the SOM is the Unified Distance Matrix, or U-matrix (Ultsch, 1993). Here, the dominant clustering structure of the observation can be seen as clearly separated areas (large distances) characterized by a homogeneous coloring. In the U-matrix, visualization of the clusters is improved by augmenting the distance matrix with additional entries (nodes) between each prototype vector and each of its neighbors. Unconventional alternatives to the U-matrix are reported by Oja et al. (2003) but not considered here.

3. CASE STUDY

To illustrate the potentialities of topological data analysis using the Self-Organizing Map, the overviewed methods are applied on a set of measurements from a full-scale process. The monitoring problem consists of modeling and analyzing the operational behaviour of an industrial deethanizer, starting from a set of online process measurements. The objective of the deethanizer (in Figure 1) is to separate ethane from the feed stream (a light naphta) while minimizing the ethane extracted from the bottom of the column (an economical constraint for the subsequent unit in the plant). Such a constraint is quantified by the maximum amount of ethane lost from the column bottom; the operational threshold is set be smaller than 2%. The

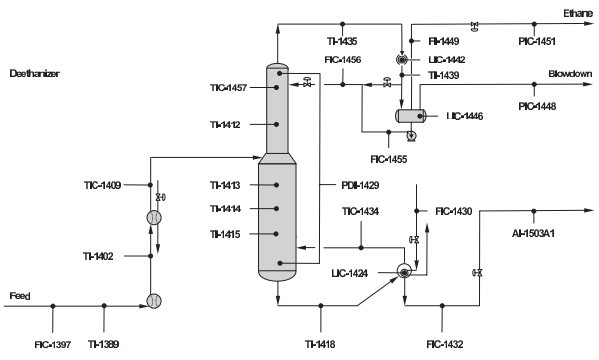


Fig. 1. Deethanizer: Simplified flowsheet.

motivation for choosing this unit is merely illustrative; in fact, the considered deethanizer offers an ample variety of behavior that reflects the operational usage; hence, an interesting groundwork for presentation and discussion.

TAG/Variable	TAG/Variable
FIC-1397/Inlet Flowrate	FIC-1430/Vapor Flowrate
TI-1389/Inlet Temp.	LIC-1424/Reboiler Level
TI-1402/Inlet Temp.	FIC-1432/Bottom Flowrate
TI-1409/Inlet Temp.	FI-1449/Distillate Flowrate
TI-1435/Top Temp.	PIC-1451/Distillate Pressure
TIC-1457/Enriching Temp.	TI-1452/Reflux Temp.
TI-1412/Enriching Temp.	FIC-1455/Bypass Flowrate
TI-1413/Exhausting Temp.	TI-1439/Condensed Temp.
TI-1414/Exhausting Temp.	LIC-1442/Top Drum Level
TI-1415/Exhausting Temp.	PIC-1448/Blowdown Pressure
TI-1418/Bottom Temp.	LIC-1446/Bottom Drum Level
FIC-1456/Reflux Flow.	AI-1503A1/Ethane Conc.
TIC-1434/Vapor Temp.	AI-1503A2/Butane Conc.
PDI-1429/Delta Pressure	

Table 1. Deethanizer: Process variables

In order to analyze the behaviour of the unit, a set of process variables was collected from the plant's distributed control system (DCS). The measurements correspond to three weeks of continuous operation in winter asset and three weeks in summer asset. The data are available as 3-minute averages and 27 process variables (in Table 1) are available for a macroscopic characterization of the unit.

In addition, there are a number of control loops in the process. Briefly, the temperature $TIC-1457$ and the vapor temperature $TIC-1434$ out of the reboiler are controlled by manipulating the reflux flow $FIC-1456$ and the steam rate $FIC-1430$ to the reboiler, respectively; with both loops cascaded with the corresponding flowrates. The distillate pressure $PIC-1451$ is controlled by the distillate flowrate $FI-1449$ and the level in the reboiler $LIC-1424$ by the bottom flowrate $FIC-1432$.

3.1 Analysis and inference

The operational objective of the column is to produce as much ethane as possible (minimizing concentration of propane from the top of the column) while satisfying the constraint on the amount of impurity from the bottom (maximum concentration of ethane in the bottom $\leq 2\%$). With respect to the loss of ethane from the bottom, such considerations led to the definition of 3 operational modes:

- a *normal* status, corresponding to the operation of the column, where the concentration of ethane is within allowable bounds (within the $1.8-2.0\%$ range)
- a *high* status, corresponding to the operation of the column, where the concentration of ethane is exceeding the allowable upper bound (above 2%)
- a *low* status, corresponding to the operation of the column, where the concentration of ethane is below the allowable lower bound (below 1.8%).

The two abnormal conditions have a direct and important economic implication. In fact, when at low status, the process is delivering a product out of specifications whereas, when at high status the product is within the specifications, but an unnecessary operational cost is observed.

To understand under which conditions such modes are experienced, in a recent study (Corona et al., unpublished) we analyzed the clustering structure of the data and visualized the operating conditions of the unit. Starting from a selection of important process variables, we expanded this

subset by incorporating an additional *dummy* indicator, specifically calculated to indicate the status. As such, the new variable was defined as to take values $+1$, -1 or 0, according to the operational status of the process. Value 0 is assigned to the normal operation, whereas values $+1$ and -1 correspond to high and low operations, respectively. Notice that the calculation of the *dummy* variable requires the availability of a real-time measurement for the ethane concentration; such a variable ($AI - 1503A1$) is presently acquired from a continuos-flow chromatograph. The subset of selected process variables augmented by the *dummy* indicator was used to calibrate a SOM over which the resulting component planes and U-matrix were analyzed; the exploration was perfomed as a direct application of the techniques discussed by Alhoniemi (2002). The study allowed us to extract the clustering structure of the data and illustrate on simple displays how it corresponds to the operational modes of the unit.

However, the delay associated with the analytical measurements of the ethane concentration from the bottom of the column can pose severe limitations to the online analysis. Moreover, the existing instrumentation setup available for the unit may benefit from a backup measurement for such an important variable. In this study, we are thus extending the analysis of the operational modes of the deethanizer, by validating the functionality of the approach when replacing the analytical measurements of ethane with online estimates. In that sense, the availability of an inference model would allow the development of a fully automated system to be implemented online in the plant's DCS.

For the purpose, a soft sensor based on the standard Multi-Layer Perceptron MLP (Haykin, 1998) with one hidden layer and sigmoidal activation functions was developed to infere the ethane concentration from the bottom. The estimates are obtained starting from the same input subset of easily measurable process variables used for the SOM and selected according to the guidelines provided by Baratti et al. (1995). The parameters of the MLP (that is, number of hidden nodes, one, and the connection weights) were optimized using the Levemberg-Marquard method and cross-validation. In Figure 2, the response of the soft sensor on a set of independent testing observations is reported for about a week of continuous operation.

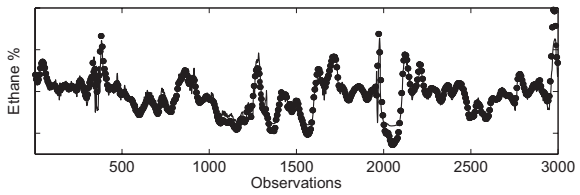


Fig. 2. Ethane concentration from the bottom: Analytical measurements ($\cdot$) and MLP estimates ($-$).

Based on the MLP estimates, a bidimensional Self-Orgaznizing Map was calibrated using only the winter data. The map consists of a hexagonal array of prototype vectors initialized in the space spanned by the eigenvectors corresponding to the two largest eigenvalues of the covariance matrix of the data. As usual, the ratio between the two largest eigenvalues was used to calculate the ratio between the two dimensions of the SOM; the resulting map consists of a 70×24 array of 15-dimensional prototype vectors, where the dimensionality of the vectors equals the number of variables used for calibration. On the SOM, we analyzed the clustering structure of the data and visualized the operating conditions of the unit using the U-matrix.

The U-matrix is based on distances between each prototype vector and its immediate neighbors. A common way to visualize it consists of an initial projection of all the distances onto a color axis and the subsequent display with colored markers between each prototype vector. On the display, areas with homogeneous coloring correspond to small within-cluster distances (recognized as clusters), whereas cluster borders are areas with homogeneous coloring but corresponding to large between-cluster distances. The use of the U-matrix in clustering the operational regimes of the deethanizer column is shown in Figure 3.

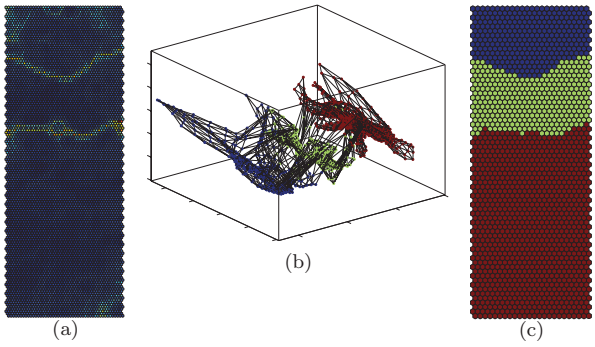
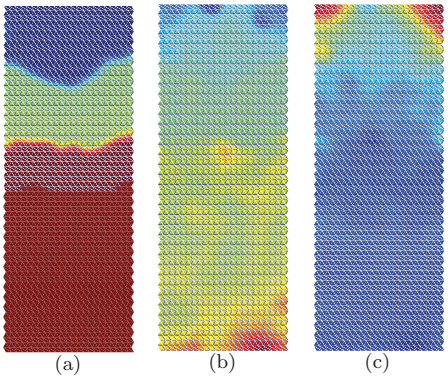


Fig. 3. The U-Matrix (a), the clustered SOM projected onto the 3D principal components space (b) and the SOM colored according to the K-means clustering (c).

In Figure 3(a), distances are depicted with dark blue colors shading toward dark red as the proximity between the prototypes decreases. The visualization permits to clearly recognize the presence of three distinct clusters of prototypes, as well as several other data substructures. An analogous visualization of the grouping is achieved by projecting the map onto a low-dimensional subsapce; in Figure 3(b), a tridimensional principal components space learned by the metric MDS. Indeed, also this visualization permits to illustrate the actual clustering structure of the process measurements and displays a good separateness also in this space of reduced dimensionality. However, to obtain a quantitative characterization of the clustering structure, the prototypes of the SOM should be regarded as a reduced data set and modeled with a standard clustering algorithm. For simplicity, we are here adopting a standard K-means algorithm coupled by the Davier-Bouldin index, a measure of cluster validity to identify an optimal number K of taxonomies from data Milligan et al. (1985). As expected, optimality was found for $K = 3$ clusters, corresponding to the modes of the unit.

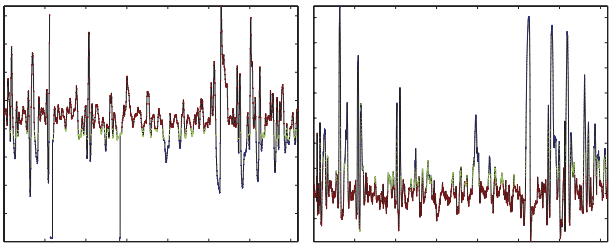
On the SOM, such clusters are located in the lower, middle and upper part of the map. After coloring the SOM according to the cluster membership obtained by using the K-means algorithms, in Figure 3(c), and comparing it with the component plane of the *dummy* variable (and equivalently, the MLP estimated ethane concentation), it is straightforward to associate the three taxonomies to the



(a) (b) (c)

Fig. 4. The component planes for the *dummy* variable (a), the estimated ethane concentration (b) and the temperature $TI-1414$ (c), with a coloring scheme that assigns blue to high values of the variables fading toward red as the values decrease. This scheme differs from the what defined for the clustering with blue and red corresponding to -1 and $+1$, respectively.

three main operational modes of the deethanizer, Figure 4. Specifically, Figure 3(c) shows the clusters on the SOM as distinct regions dyed in blue, green and red with a cloring scheme that assignes those colors to the operational modes ($+1$, 0 and -1, respectively). As expected, a similar structure is also retrieved from the component plane for the *dummy* variable, Figure 4(a). Though apparently less evident, the same structuring is retrieved from the component planes of the estimated ethane concentration (Figure 4(b)) and one of the temperatures in the exhausting section of the dethanizer; namely, $TI-1414$ in 4(c). Looking for similar patterns in similar positions in such components planes allows the visualization of a neat dependence between the ethane composition and such temperature indicator. Such pair of variables shows near identical but reversed component planes, thus highlighting the inherent inverse correlation that exists between them.



(a) AI-1503A1 (b) TI-1414

Fig. 5. The colored time series (3 winter weeks) for the ethane concentration $AI-1503A1$ (a) and the temperature in the enriching section $TI-1414$ (b). The actual values of the variables could not be reported because of the confidentiality agreement.

Information about this dependence can be further enhanced by applying the coloring scheme resulting from clustering directly to the original observations in the time domain. In fact, all points can be dyed using the cluster color of the corresponding Best Matching Unit, as in Figure 5. The figure shows how $TI-1414$ is mostly responsible for the transition between the aformentioned operational

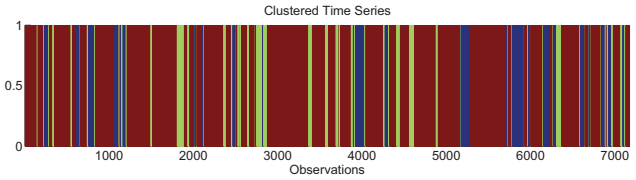
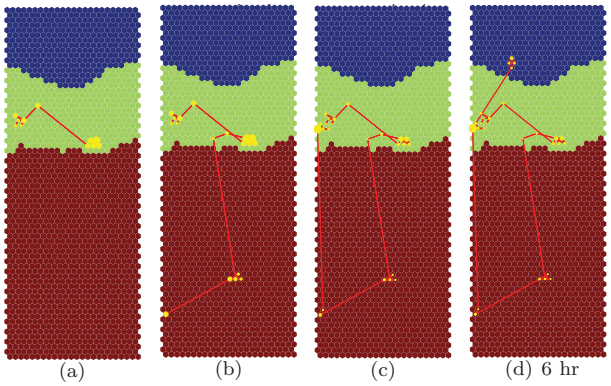


Fig. 6. The temporal evolution of the winter operational modes colored according to the SOM clustering.

conditions. The correspondence with the ethane concentration is observed as clear banded regions and indicates that, in order to maintain the column at optimality (withing the $1.8-2.0\%$ range of ethane from the bottom), such a temperature should be controlled (possibily, within the $52-55°C$ range). A possible variable to manipulate is the steam flowrate $FIC-1430$. However, such a variable is not used in the present control scheme and induces an overall 85% of off-spec operation of the unit, during the given winter period. Such information is obtained by calculating the number of point measurements that falls outside the normality conditions over the total count and pictorially depicted also as clustered time series (in Figure 6).

So far, we have restricted the analysis only to the measurements observed under winter asset. However, it is also possible to directly use the calibrated SOM as a reference model for new and unseen observations; in our setting, the three weeks of data corresponding to the summer operation of the deethanizer column. To validate this idea, the winter SOM was used to explore the behaviour of the deethanizer under summer asset. Again, the summer measurements from $AI-1503A1$ were replaced by the estimates from the soft sensor. The analysis was accomplished by initially projecting the new data onto the calibrated SOM, being the mapping based on a nearest neighbor criterion between the new sample vectors and the prototype vectors of the SOM. In this respect, novelty detection using the SOM is based on finding the BMU. Once the mapping is completed, the inspection is performed for the new data.

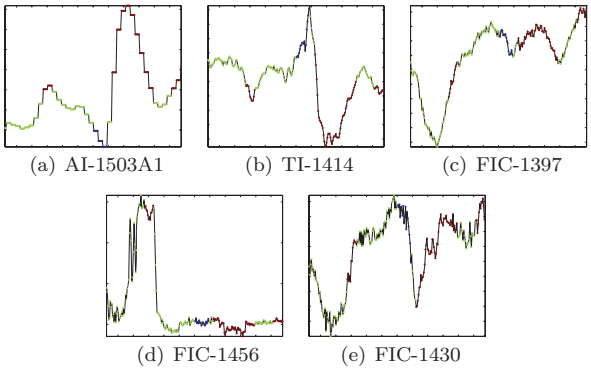


(a) (b) (c) (d) 6 hr

Fig. 7. Trajectory of a selection the summer observations (approximatively, 6hr) displayed on the winter SOM.

The results in extrapolation are presented by illustrating another technique for visualization on the SOM. The approach allows to follow operational changes in the process and tries to provide a simple display for identifying reasons of specific behaviors. For the purpose, the map calibrated on the winter data can be enhanced by the inclusion of

the summer point trajectories followed by the process. The trajectory permits to intuitively indicate the current mode of the process and observe how it has been reached. In Figure 7, the process trajectory is sequentially reported for a small time window corresponding to six hours of continuous summer operation of the deethanizer. The process trajectory on the SOM's domain passes through all the BMUs of each new data vector and it is shown as red line connecting the visited prototypes (the nodes are marked as yellow dots and thicken with the count of visits).



(a) AI-1503A1 (b) TI-1414 (c) FIC-1397

(d) FIC-1456 (e) FIC-1430

Fig. 8. Status transitions on the time domain (approximatively, 6hr), for a set of relevant process variables.

Following the temporal evolution from Figure 7 and 8, the diagrams show a process that is initially operated in the green area, or *normal* condition (as for the ethane in the bottom and reference temperature). As the process has moved further in time, new prototype vectors are visited and added to the trajectory until the column eventually leaves the normality region and crosses the boundary towards the region of high ethane composition (in red). In a similar fashion, all the process variables changed coloring to match the visited modes allowing to appreciate that the change in the operation was mainly due to an abrupt change in the feed flowrate ($FI-1397$), in Figure 8(c), and possibly its composition. In turns, the variation triggered the action on steam to reboiler flowrate ($FIC-1430$), in Figure 8(e), as well as the reflux to control the top temperature ($FIC-1456$), in Figure 8(d). The events initiated a sequence of oscillations around normality that could be reestablished only after several hours.

4. CONCLUSIONS

In this work, we implemented and discussed a strategy to model, visualize and analyze the information encoded in industrial process data. In particular, the proposed strategy was applied to a full-scale distillation column.

From a methodological point of view, the process monitoring problem was casted in a topological framework by using the Self-Organizing-Map. On the SOM, the identification of the process modes was approached as a clustering task rather than classification; that is, in an unsupervised rather than supervised fashion. Moreover, in order to overcome the limitations associated with the time delay and costs of the analytical instrumentation, a software sensor based on a Multi-Layer Perceptron was developed to infer a primary process variable, thus favoring the possibility to directly use such a strategy also for online monitoring.

The application allowed the definition of simple displays capable to present meaningful information on the actual state of the process and also suggested an alternative control strategy for maintaining the unit in normal conditions.

ACKNOWLEDGEMENTS

J. Romagnoli kindly acknowledges Regione Sardegna for the support through the program *Visiting Professor 2008*.

REFERENCES

E. Alhoniemi. Unsupervised pattern recognition methods for exploratory analysis of industrial process data. *Doctoral Dissertation*, Lab. of Computer and Information Science. Helsinki University of Technology, Finland, 2002.

R. Baratti, G. Vacca, and A. Servida. Neural network modeling of distillation columns. *Hydrocarbon Processing*, 74:35–38, 1995.

F. Corona, M. Mulas, R. Baratti, and J.A. Romagnoli. On the topological analysis of industrial process data. PSE 2009 International Symposium on Process Systems Engineering, to appear.

T.F. Cox, and M.A.A. Cox. *Multidimensional scaling, Second edition*. Chapman & Hall, 2000.

P. Demartines, and J. Herault. Curvilinear component analysis: a self-organizing neural network for nonlinear mapping of data sets. *IEEE Transaction on Neural Networks*, 8:148–154, 1997.

S. Haykin. *Neural Networks: A Comprehensive Foundation, Second Edition*. Prentice Hall, 1998.

I.T. Jolliffe. *Principal Components Analysis, Second edition*. Springer, 2002.

S. Kaski. Data exploration using Self-Organizing Maps. *Doctoral Dissertation*, Lab. of Computer and Information Science. Helsinki University of Technology, Finland, 1997.

T. Kohonen. *Self Organizing Maps, Second edition*. Springer, 2001.

A. Hartigan, and M.A.A. Wong. K-means clustering algorithm. *Applied Statistics*,28:100–108, 1979.

S. Laine. Using Visualization, Variable selection and feature extraction to learn from industrial data. *Doctoral Dissertation*, Lab. of Computer and Information Science. Helsinki University of Technology, Finland, 2003.

P. Lloyd. Least squares quantization in PCM. *IEEE Transaction on Information Theory*, 28:129–137, 1982.

G.W. Milligan, and M.C. Cooper. An examination of procedures for determining the number of clusters in a dataset. *Psychometrika*, 50:159–179, 1985.

M. Oja, S. Kaski, and T. Kohonen. Bibliography of Self-Organizing Map (SOM) papers: 1998-2001 addendum. *Neural Computing Surveys*, 3:1–156, 2003.

A. Ultsch. Self-organizing neural networks for visualization and classification. *Information and Classification*, 307–313. Springer, 1993

J. Vesanto. Data exploration process based on the Self-Organizing Map. *Doctoral Dissertation*, Lab. of Computer and Information Science. Helsinki University of Technology, Finland, 2002.

Stiction Identification in Nonlinear Process Control Loops

U. Nallasivam [*] **B. Srinivasan,** [**] **R. Rengaswamy** [***]

[*] *Dept. of Chemical Engineering, Clarkson University, Potsdam, NY, US 13699. (e-mail: nallasu@clarkson.edu).*
[**] *Dept. of Chemical Engineering, Texas Tech University, Lubbock, Texas, US 79409. (e-mail: babji.srinivasan@ttu.edu)*
[***] *Dept. of Chemical Engineering, Texas Tech University, Lubbock, Texas, US 79409. (e-mail: raghu.rengasamy@ttu.edu)*

Abstract: Nearly 20-30% of all process control loops oscillate due to stiction and lead to loss of productivity. Thus, the detection and quantification of stiction in control valves using just the raw operating data is an important component of any automated controller performance monitoring application. Many techniques have been proposed for the detection and quantification of stiction. Pattern based identification approaches use unique shapes of the PV and OP data to identify stiction. Other approaches that include some measure of nonlinearity index have also been used to identify stiction. A solution technique for stiction detection in nonlinear processes with known process models is also available. In this paper, one possible approach to detect stiction in nonlinear process control loops with unknown process models is discussed.

1. INTRODUCTION

Research on developing automated controller performance monitoring systems has been increasing in the past decade. Control strategies such as Model Predictive Control (MPC) or other supervisory control are crucial for optimization of process operations. Performance gains from such advanced control techniques depend on how effectively the lowest control elements in the control strategy track the desired set points. A spate of surveys on the performance of control loops [Bialkowski, 1993, Ender, 1993, Entech, 2005, Desborough and Miller, 2001] indicate that a majority of control loops in processing industries perform poorly. Performance demographics of 26,000 PID controllers collected across a wide variety of processing industries in a two year time span indicate that the performance of 16% of the loops can be classified as excellent, 16% as acceptable, 22% as fair, 10% as poor, and the remaining 36% are in open loop [Desborough and Miller, 2001]. The impact of this has to be seen from the fact that PID is the dominant control algorithm in the industry accounting for 97% of the regulatory loops [Desborough, 2003]. This has led to increasing interest in automated Controller Performance Assessment (CPA) tools in recent years. The three major reasons for deterioration of control performance are: badly tuned controllers, oscillating load disturbances, or nonlinearities in control valves. 20% to 30% of all control loops oscillate due to valve problems caused by static friction or hysteresis [Bialkowski, 1993, Miller, 2000] resulting in performance deterioration. It was found that over 80% of all valves adjusted by Entech Control Engineering failed dynamic performance standards due to stiction, backlash or oversized design. Thus the task of detecting stiction or other nonlinearities in valves from routine operating data is a challenging task and is an important component in any CPA suite.

⋆ Corresponding Author: R. Rengaswamy, Dept. of Chemical Engineering, Texas Tech University, Lubbock, Texas, US 79409. Adjunct Faculty, Dept. of Chemical Engineering, Clarkson University.

2. PROBLEM DEFINITION

A typical process control loop with stiction in the control valve can best be depicted as shown in the Figure 1. As seen, the stiction precedes the control valve dynamics and the process transfer function also includes the valve dynamics. The fundamental problem that is being solved in this paper is one of identifying the root cause of oscillation in the process variable (PV) as being due to either stiction or external oscillations. In this work, the focus is on a model-based solution approach to this problem. There are solutions for stiction detection based on the analysis of the input-output data such as the shape based analysis proposed by Rengaswamy et al. [2001], Srinivasan et al. [2005a] and higher order statistics based approach proposed by Choudhury et al. [2004]. Most of these approaches rely on the process being linear. For non-linear process Nallasivam and Rengaswamy [2008] proposed a solution strategy that works when the process model is known. However, there is no work on detecting stiction in nonlinear control loops when the process model is not known.

Previous attempts at quantifying stiction were mostly based on measures developed from the data characteristics such as the span of controller output (OP) data, apparent stiction, maximum width of the ellipse fitted by PV-OP plot etc. The first attempt at quantifying stiction through a joint identification procedure was by Srinivasan et al. [2005b]. Srinivasan et al. [2005b] proposed a model-based approach and solved this problem for a linear process. Their approach is based on the identification of a Hammerstein model of the system comprising of the sticky valve and the process (see Figure 1(b)). The identification of the linear dynamics is decoupled from the nonlinear element. The decoupling between the nonlinear and the linear component is achieved by an iterative procedure. The solution proposed in Srinivasan et al. [2005b] is shown in Figure 2. Several stiction quantification attempts based on this approach have started to appear. A similar approach but

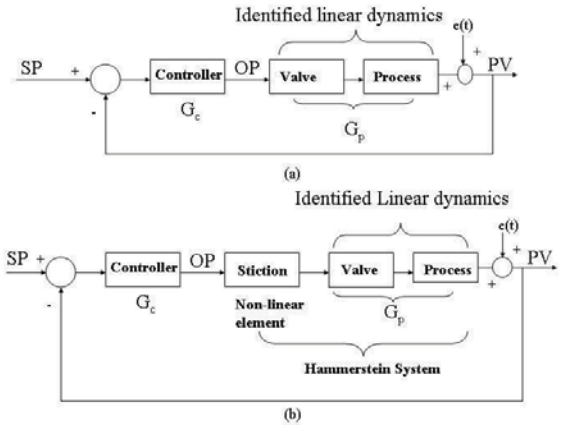


Fig. 1. (a) General process control loop, (b) Process control loop with stiction element

with a two parameter model to quantify stiction is discussed in Choudhury et al. [2008]. Another work using a Hammerstein ID approach with a two parameter model can be found in Jelali [2008]. The difference between Choudhury et al. [2008] and Jelali [2008] seem to be that while a gird search, similar to Srinivasan et al. [2005b], is used in Choudhury et al. [2008], genetic algorithms (GAs) are used to identify the stiction parameters in Jelali [2008]. However, all these methods assume that the process is linear. Nallasivam and Rengaswamy [2008] have shown that these approaches fail if the underlying process is nonlinear and solved this problem for the nonlinear case when the process model is known. The present work is on detecting stiction in nonlinear control loops when the process model is unknown.

Figure 3 depicts the control loop that is being addressed in this work. From this figure,

$$\begin{aligned} y &= y_p + y_d \\ y &= N(u) + y_d \\ y &= N(V(v)) + y_d \end{aligned} \tag{1}$$

where y is the measured process variable pv, which includes the process component y_p and the disturbance component y_d, which are additive. N is the non linear process transfer function and u is the valve output, which might not be a measured variable. The valve output u is a function (V) of the op (v) dictated by the stiction phenomenon. In this paper, the detection, quantification and isolation of stiction from external disturbances for the system given in equation 1 is addressed.

$$x(t) = \{ \begin{matrix} x(t-1) & if\ |u(t) - x(t-1)| \le d, \\ u(t) & otherwise \end{matrix} \tag{2}$$

3. SOLUTION APPROACH

A single parameter stiction model is given by equation 2. In this model, the value of the parameter d goes to zero when stiction is absent in the valve. Thus a non-zero value for this parameter d indicates the presence of stiction and also quantifies the stiction level. The estimation of this parameter is achieved by decoupling the stiction parameter estimation from the estimation of the process dynamics. This is achieved by

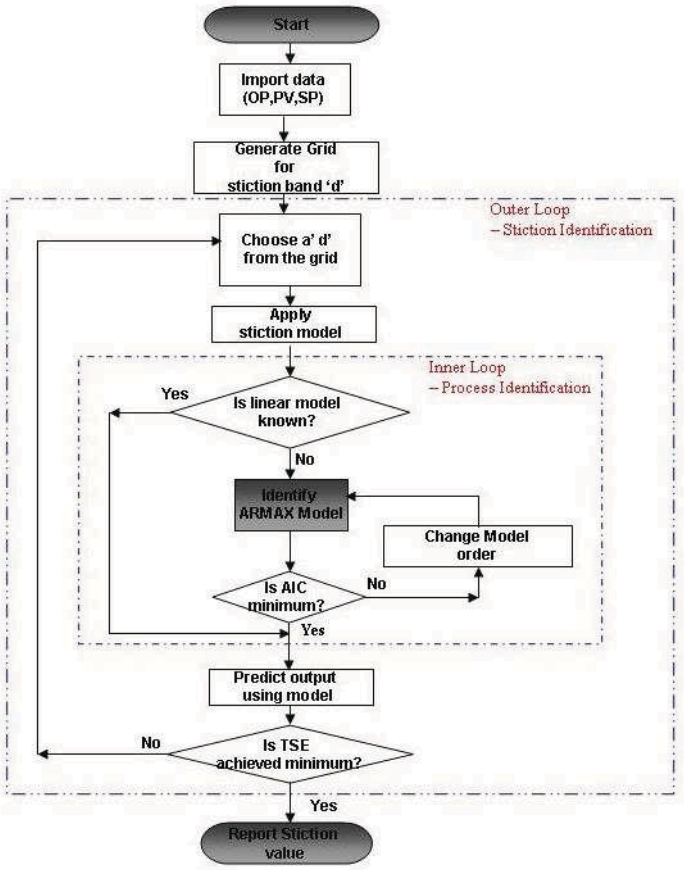


Fig. 2. Solution algorithm proposed by Srinivasan et al. [2005b]

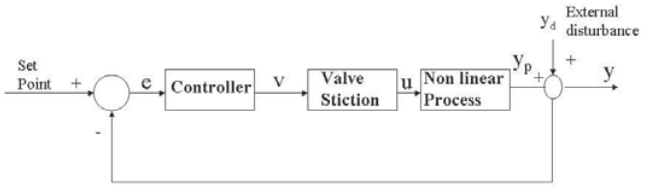


Fig. 3. Nonlinear control loop in presence of stiction

an iterative process in which a value for d is assumed in an outer loop and the best fit model for the remaining dynamics in the inner loop is identified. From Figure 3, since the controller parameters θ_c are known, v (op) can be calculated from y. Using the equation 2 for a given selected value of d, u can be calculated. Thus the identification problem becomes,

$$\begin{aligned} y &= y_p + y_d \\ y &= N(u) + y_d \end{aligned}$$

Since the process model is not known, by considering y_d as a moving average process, we can write

$$A(q)y(t) = B(u,q) + C(q)e(t) \tag{3}$$

where

$$A(q) = [1 + a_1 q^{-1} + ... + a_{n_a} q^{-n_a}]$$
$$C(q) = [1 + c_1 q^{-1} + ... + c_{n_c} q^{-n_c}]$$

$B(u,q)$ represents a general nonlinear process. As before u is known based on the actual output, the controller parameters and the assumed d value. A predictor form can be obtained for the system given by equation 3. In the linear model case, this will lead to a pseudolinear regression problem. One approach to retain the pseudolinear regression framework in the nonlinear case would be to parameterize the nonlinear function using a N^{th} order discrete Volterra series approximation as below

$$B(u,q) = \sum_{n=1}^{N} \sum_{i_1=1}^{M_1} ... \sum_{i_n=1}^{M_n} h_n(i_1,...,i_n) \, q^{-i_1} u(k) ... q^{-i_n} u(k)$$

With this expression, equation 3 now represents a Volterra MA model. By considering only the first and second order terms in the above Volterra series,

$$B(u,q) = \sum_{i=1}^{n_b} h_1(i) q^{-i} u(k) + \sum_{i=1}^{n_b} \sum_{j=1}^{n_b} h_2(i,j) q^{-i} u(k) q^{-j} u(k)$$

Now the predictor for equation 3 can be derived as

$$y(k/k-1) = B(u,q) + [1 - A(q)]y(k) + [C(q) - 1]e(k)$$

which is

$$y(k/k-1) = \sum_{i=1}^{n_b} h_1(i) q^{-i} u(k) + \sum_{i=1}^{n_b} \sum_{j=1}^{n_b} h_2(i,j) q^{-i} u(k) q^{-j} u(k) - a_1 y(k-1) - a_2 y(k-2) - ... - a_{n_a} y(k-n_a) + c_1 e(k-1) + c_2 e(k-2) + ... + c_{n_c} y(k-n_c)$$

When this predictor is applied to n samples, one would get n equations which results in the following equation in the matrix form

$$Y = XH$$

This equation can be solved iteratively till the solution converges for a given given selected model order of n_a, n_b and n_c using the following relationship.

$$H = [X^T X]^{-1} X^T Y$$

Based on this second order approximation of the Volterra series, an approach similar to the one that was used by Nallasivam and Rengaswamy [2008] for the known model case can be followed. However, now the model parameters and the MA process parameters have to be jointly estimated and evaluated through the AIC criteria. The overall best fit could then be chosen based on the d parameter that results in the minimum TSE. This approach is shown in Figure 4.

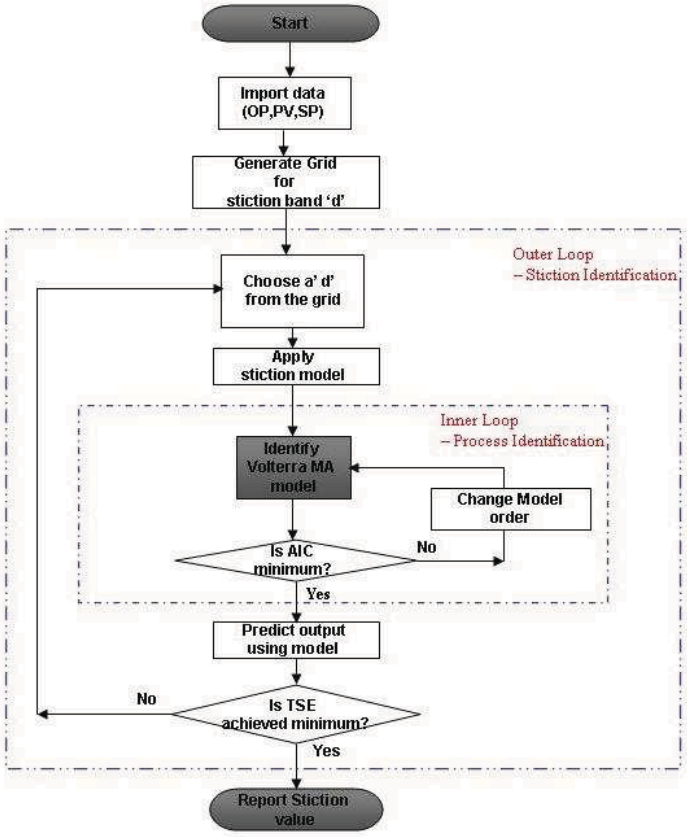


Fig. 4. Proposed approach

4. CASE STUDY

In this case study, a nonlinear polymerization reactor process from Doyle et al. [1995] is used. In this nonlinear process a polymerization reaction takes place in a jacketed CSTR where the controlled variable is the number-average molecular weight and the manipulated variable is the volumetric flowrate of the initiator. A second-order Volterra model in the frequency domain as given below describes this non-linear process.

$$P_1 = c_1^T (sI - A_{11})^{-1} b_1$$
$$P_2 = c^T [(s_1 + s_2)I - A]^{-1} N (s_1 - A)^{-1} b \tag{4}$$

Details on the matrices c, A, N, b can be found in Doyle et al. [1995].

4.1 Data for testing of the solution approach

This case study is used to demonstrate the effectiveness of the proposed solution approach in three different scenarios for stiction detection. These are:

(a) No stiction case
(b) Stiction alone case
(c) Stiction and external oscillation case

Three datasets were generated by using equation 4 as the nonlinear process in Figure 3 to address all the above three scenarios. A PI controller with $K_p = 0.3$, $T_i = 1.0$ was used. Data were

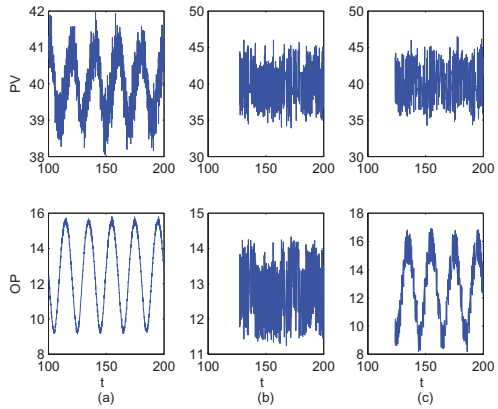


Fig. 5. Data for (a) No stiction (b) Stiction alone (c) Stiction and external oscillation

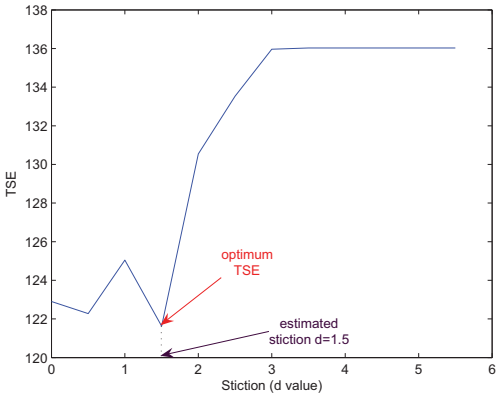


Fig. 6. Result for the approach of Srinivasan et al. [2005b]

simulated for scenario (a) using an external sine oscillation disturbance of amplitude 20 at a frequency of $0.3142 rad/sec$ as y_d. For scenario (b), a stiction value of $d = 1.5$ was used. For scenario (c), both the sine oscillation of scenario (i) and a stiction value of $d = 1.5$ were used. The data that are generated are shown in Figure 5.

4.2 Discussion on the existing approaches

The existing techniques based nonlinearity detection as in Choudhury et al. [2004] and qualitative pattern matching approaches as the one proposed in Rengaswamy et al. [2001] will not work for this dataset. As shown in Nallasivam and Rengaswamy [2008], the model-based approach proposed by Srinivasan et al. [2005b] is also not likely to work for this dataset. To verify this, the data shown in Figure 5(a) for the no stiction case is tested using the approach suggested in Srinivasan et al. [2005b] (approach shown in Figure 2). The resulting d vs TSE plot is shown in Figure 6. As expected, the value of d is incorrectly identified. In other words, stiction is detected where it is not actually present.

5. RESULTS

The dataset (Figure 5(a)), where the approach of Srinivasan et al. [2005b] failed is used to test the performance of the

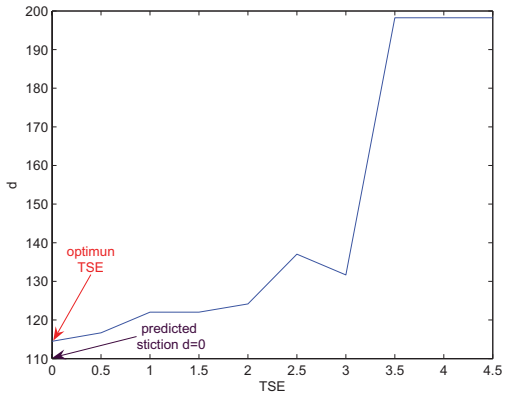


Fig. 7. Result for the no stiction case

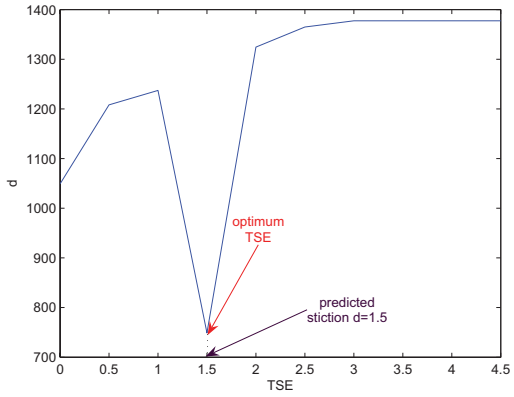


Fig. 8. Result for the stiction alone case

proposed approach shown in Figure 4. Figure 7 shows the result and as seen, it is clear that the scenario is correctly diagnosed as being a no stiction case. The minimum TSE is achieved at $d = 0$.

The dataset for the other two scenarios (Figures 5(b) and 5(c)) are also tested using this proposed approach. The results are shown in Figures 8-9. It can be seen from Figure 8, the case of stiction is also correctly identified with an accurate estimation of the stiction level. The more difficult third scenario is where both stiction and an external oscillating disturbance are present, with the process being nonlinear. The result for this case is shown in Figure 9. In this case also, not only is stiction detected but the magnitude of stiction is also accurately estimated. From these observations, it clear that the proposed solution approach works well in detecting and isolating the root cause of oscillation in nonlinear SISO loops.

6. VALIDATION USING DATA FROM PHYSICAL STICTION MODEL

The aim of this section is to verify how the proposed approach works when process data is generated by considering physical stiction model instead of single parameter stiction model. The physical stiction model that was used for this simulation is the same as the one used by Srinivasan et al. [2008]. The various parameters that were used for the physical stiction model are given in Table 1.

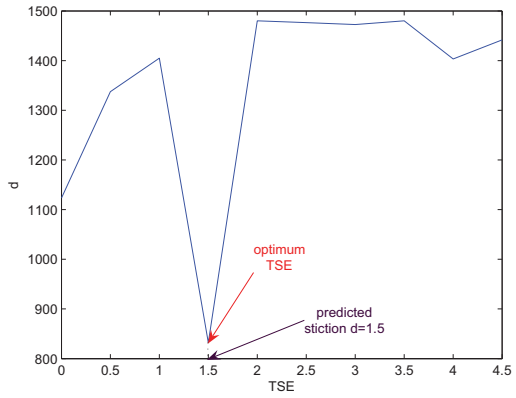


Fig. 9. Result for the case of both stiction and an external oscillating disturbance case

Table 1. Valve Parameters

Parameter	Description	Value
P	Applied Actuator Pressure	psi
A	Effective Diaphragm Surface area	100 in^2
m	Mass of Stem and Plug	3 lb
K	Spring rate	300 lbf/in
b	Viscous coefficient	0.15 lb/s
F_c	Coulomb friction	24 lbf
F_s	Static friction	34 lbf
v_s	Stribeck constant	0.01 in/s

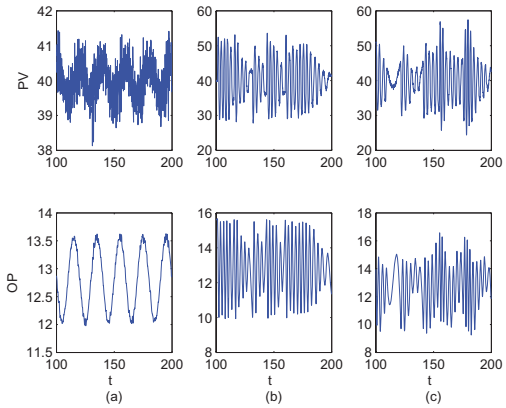


Fig. 10. Data set 2 for (a) No stiction (b) Stiction alone (c) Stiction and external oscillation

Data were simulated for scenarios (a) and (c) using an external sine oscillation disturbance of amplitude 5 at a frequency of $0.3142 rad/sec$ as y_d. A PI controller with $K_p = 0.01$, $T_i = 0.5$ was used for this data generation. The data that are generated are shown in Figure 10.

This data are tested using the proposed approach and the results are given in the Table 2. As seen, the absence or presence of stiction is predicted correctly in all the scenarios as indicated by a zero or non zero d value respectively. However the d parameter estimated for scenarios b and c are not the same as one would expect because of the use of the physical stiction model in generating the data. The reason for this is that the single parameter stiction model used in the detection algorithm is only an approximation of the physical stiction model. Nonetheless, stiction detection is not compromised.

Table 2. Validation Results

Scenario	Predicted stiction
No stiction case	d=0
Stiction alone case	d=3
Stiction and disturbance case	d=0.5

7. DISCUSSION

The proposed approach takes advantage of the fact that the stiction nonlinearity is discontinuous, whereas the process transfer function is continuous for stiction detection. It was shown that it might be possible to detect and isolate stiction in some cases in nonlinear SISO control loops when the process model is not known. However, extensive studies are needed before any definite conclusions can be drawn. There are several possible extensions to the proposed approach. The obvious ones include the use of two parameter stiction model for stiction quantification, the use of optimization algorithms such as GA for estimating the stiction parameters and validation with industrial data. Also, further theoretical work is needed to formalize the approach proposed in this paper.

8. CONCLUSIONS AND FUTURE WORK

In this paper, the problem of detection of stiction and isolation of stiction from external oscillations in nonlinear process control loops was addressed for the unknown model case. While Nallasivam and Rengaswamy [2008] have demonstrated the solution strategy for known nonlinear model case, almost no work exists in the case of unknown nonlinear processes. A solution approach for the unknown model case was proposed. The advantages and the limitations of the proposed approach were discussed.

It is essential to analyze the theoretical basis of the proposed method for using Volterra models. In addition, it would be interesting to study the use of Volterra second-order models to higher order nonlinear systems or linear systems. The former results in under-modeling of the original process while the latter results in over-modeling of the underlying linear process. We are in the process of developing a theoretical basis to analyze these interesting phenomena [Nallasivam et al. [2009]]. In future, the efficacy of the proposed approach along with the underlying theory needs to be further validated with other examples for different types of disturbances and different stiction models.

ACKNOWLEDGEMENTS

The authors thank the National Science Foundation (USA) for financial support for this work through the Grant CBET-0553992.

REFERENCES

W. L. Bialkowski. Dreams versus reality: A view from both sides of the gap. *Pulp and Paper Canada.*, 94(11):19–27, 1993.

M. A. A. S. Choudhury, M. Jain, and S. L. Shah. Stiction - definition, modelling, detection and quantification. *Journal of Process Control*, 18:232–243, 2008.

M. A. A. S. Choudhury, S. L. Shah, and N. F. Thornhill. Diagnosis of poor control loop performance using higher order statistics. *Automatica*, 40(10):1719–1728, 2004.

L. D. Desborough. Control loop economics. *Honeywell Proprietary report*, 2003.

L. D. Desborough and R. M. Miller. Increasing customer value of industrial control performance monitoring – honeywell's experience. Arizona, USA, 2001. CPC-VI.

F. J. Doyle, B. A. Ogunnaike, and R. K. Pearson. Nonlinear model-based control using second-order volterra models. *Automatica*, 31(5):697–714, 1995.

D. B. Ender. Process control performance: Not as good as you think. *Control Engineering*, 9:180–190, 1993.

Entech. Control valve dynamic specification. version 3.0. Entech Control, Division of Emerson Electric Canada Limited, Canada, Toronto, 2005.

M. Jelali. Estimation of valve stiction in control loops using separable least-squares and global search algorithms. *article in press, Journal of Process Control*, 2008.

R. M. Miller. Loop scout regulatory control performance study. *Honeywell*, unpublished report, 2000.

U. Nallasivam and R. Rengaswamy. Blind identification of stiction in nonlinear process. Seattle, USA, 2008. American Control Conference.

U. Nallasivam, B. Srinivasan, and R. Rengaswamy. On the detection of stiction in closed-loop systems. *Automatica*, to be submitted, 2009.

R. Rengaswamy, T. Hägglund, and V. Venkatasubramanian. A qualitative shape analysis formalism for monitoring control loop performance. *Engineering Applications of Arti.cial Intelligence*, 14:23–33, 2001.

R. Srinivasan, R. Rengaswamy, and R. M. Miller. Control loop performance assessment 1: A qualitative pattern matching approach for stiction diagnosis. *Industrial and Engg. Chemistry Research*, 44:6708–18, 2005a.

R. Srinivasan, R. Rengaswamy, U. Nallasivam, and V. Rajavelu. Issues in modelling stiction in process control valves. Seattle, USA, 2008. American Control Conference.

R. Srinivasan, R. Rengaswamy, S. Narasimhan, and R. M. Miller. Control loop performance assessment 2: Hammerstein model approach for stiction diagnosis. *Industrial and Engg. Chemistry Research*, 44:6719–28, 2005b.

Stochastic dynamical nonlinear behavior analysis of a class of single-state CSTRs

S. Tronci*, M. Grosso*, J. Alvarez*[+] and R. Baratti*

*Dipartimento di Ingegneria Chimica e Materiali, Università degli Studi di Cagliari, I-9123 Cagliari, Italy
(Tel: +39-0706755056; e-mail: tronci;baratti;grosso@ dicm.unica.it)
[+] On leave from Departamento de Ingenieria de Procesos e Hidraulica, Universidad Autonoma Metropolitana - Iztapalapa, 09340 Mexico D.F., Mexico (e-mail:jac@xanum.uam.mx)

Abstract: Motivated by the need of developing stochastic nonlinear model-based methods to characterize uncertainty for chemical process estimation, control, identification, and experiment design purposes, in this paper the problem of characterizing the global dynamics of single-state nonlinear stochastic system is addressed. An isothermal CSTR with Langmuir-Hinshelwood kinetics is considered as representative example with steady state multiplicity. The dynamics of the state probability distribution function (PDF) is modeled within a Fokker-Planck's (FP) global nonlinear framework, on the basis of FP's partial differential equation (PDE) driven by initial state and exogenous uncertainty. A correspondence between global nonlinear deterministic (stability, multiplicity and bifurcation) and stochastic (PDF stationary solution and mono/multimodality) characteristics is identified, enabling the interpretation of tunneling-like stationary-to-stationary PDF transitions, and the introduction of a bifurcation diagram with the consideration of stochastic features in the context of the CSTR case example.

Keywords: isothermal CSTR, multistability, nonlinear system, stochastic model, Fokker-Planck

1. INTRODUCTION

The study of stochastic nonlinear systems is motivated by the need of characterizing the effect of model uncertainty for model-based applications such as chemical process modeling, system identification, experiment design, estimation and control purposes, process safety assessment. While the deterministic approaches for nonlinear chemical processes are a rather mature field, the development of nonlinear stochastic approaches lags far behind. Deterministic descriptions suffice for chemical processes described by nonlinear models over the neighborhood of a steady-state (or nominal motion) for a continuous (or batch) process, but the same cannot be said for processes which evolve over ample (nonlocal) state-space domains, where nonlinearities become significant, and consequently, the inexorable presence of uncertainty due to measurement and modeling errors and its effect on the stability, observability and controllability features must be regarded within a stochastic global nonlinear framework. In chemical processes, the combination of measurement errors with high-frequency unmodeled dynamics manifests itself as random-like uncertainty, which imposes limits of estimation and control behavior.

In most of previous studies in chemical process engineering, the issue of uncertainty characterization has been performed with the so-called model sensitivity analysis with respect to initial values and/or parameters (Morbidelli and Varma, 1989, Dutta *et al.*, 2001), on the basis of a linear model truncation. The drawback of this approach is that it does not allow the assessment of the combined effect of the uncertainties caused by the neglected nonlinear dynamics, which manifest itself when testing or implementing the model with the data generated by the actual nonlinear process (Horenko *et al.*, 2005).

In the nonlinear systems theory field, there are rather well established approaches to address the model uncertainty problem for multi-state nonlinear processes, with rigorous probability distribution function (PDF) evolution descriptions in terms of a set of Fokker-Planck (FP) partial differential equations (Risken, 1996). In fact, the nonlinear EKF estimator design can be seen as a second-order statistics approximation of the FP equation approach. However, in spite of being the EKF the most widely used estimation technique in chemical process systems engineering, its employment for uncertainty assessment purposes has been rather limited, and the consideration of the full nonlinear statistics FP equation approach has been circumscribed to a rather limited set of studies.

While the rigorous FP equation approach has been successfully applied in a diversity of problems in applied science, including physics, medical sciences (Mei *et al.*, 2004; Lo, 2007), biology (Soboleva and Pleasants, 2003; Huang *et al.*, 2008) and electronic circuits (Hanggi and Jung, 1988), in the chemical process systems engineering field only a few chemical reactor studies have been performed according to the FP equation approach. In a pioneering work, Pell and Aris (1969) studied the local-stochastic behavior of a

chemical reactor on the basis of a linear model truncation. In spite of the limited nature of the local results, recognized by the authors themselves, this work evidenced the benefit and possibilities of modeling the presence of random fluctuations within a stochastic framework. Later, Rao *et al.* (1974) addressed the same problem with a numerical algorithm to solve the associated nonlinear equation drawing nonlocal results and establishing that the linearization approach breaks down when the system is close to a saddle-node bifurcation. In a subsequent study, Ratto (1998) applied the FP equation approach to the linearization of a stable closed-loop reactor with PI temperature control subjected to measurement noise, sufficiently away from the possibility of Hopf bifurcations (whose consideration is a central point of the present study). This study evidenced the advantages of the FP equation-based theoretical approach (with quasi-analytical solutions), with respect to Monte Carlo methods (Ratto and Paladino 2000, Paladino and Ratto 2000, Sherer and Ramkrishna, 2008; Hauptmanns, 2008).

In the context of a combustion engineering science problem, Oberlack *et al.* (2000) studied the stationary solution of the FP equation associated to a multistable homogeneous adiabatic flow reactor described by a one-dimensional deterministic system. In spite of having addressed only the steady-state aspect of the problem, this study further evidenced the capabilities and possibilities of the FP equation-based approach to tackle the chemical reactor stochastic modeling problem. These considerations on the employment of the FP equation-based approach for the treatment of dynamical nonlinear systems, in general, and of chemical reactor, in particular, motivate the present study on the global-stochastic dynamical behavior of chemical reactors with emphasis on: the presence of multistability, transient behavior and the connection between deterministic and stochastic modeling approaches.

As an inductive step towards the development of nonlocal, global, nonlinear stochastic uncertainty characterization methodology, in this work the problem of characterizing the concentration stochastic dynamical behavior of single-state nonlinear isothermal CSTR with Langmuir-Hinshelwood kinetics as representative case example with multistability phenomena has been addressed. The problem is treated within a global-nonlinear framework by combining deterministic multiplicity and bifurcation analysis tools with a FP equation-based stochastic behavior characterization, in the light of the particular system characteristics. The stochastic dynamical behavior is studied by looking at the response solution of the dynamic FP partial differential equation (PDE) to: (i) initial state uncertainty and (ii) modeling error described as a white noise exogenous input injection. As a result, a correspondence between stochastic features (mono or multimodality, potential, quasi-stability, and escape time) and deterministic features (stability, multiplicity and bifurcation) is established, enabling a better understanding of the nonlinear stochastic behavior and opening the possibility of extending the approach to multi-state chemical processes.

2. THE STOCHASTIC MODEL

Consider the single-state (x) nonlinear stochastic dynamical system:

$$\dot{x} = f[x, u(t)] + w(t), \quad x(0) = x_o, \quad x \in X = [0, \infty) \qquad w(t) \sim N[0, q(x)] \qquad (1)$$

with exogenous deterministic input u, and driven by input uncertainty modeled as white noise with intensity $q(x)$. In the absence of noise, with $w(t) = 0$, the (single or multiple) steady-states satisfy, for the nominal input $\bar{u}$, the static-algebraic equation $f(\bar{x}, \bar{u}) = 0$. Due to the nonlinearity of $f(x)$, the deterministic system (i.e. when $w(t)=0$) can show structural instability, meaning the existence of steady-state bifurcation points as system parameters or inputs are varied. In the one-dimensional case, the more generic bifurcation is the saddle-node, which may imply the presence of multistability regions. This means that the deterministic system reaches one of the stable equilibrium points, depending on initial conditions and system input (Wiggins, 1990). Assuming the noise intensity $q(x)$ is constant for a fixed value of the input, $u(t) = \bar{u}$, the dynamics of the concentration (normalized) probability density function (PDF) $p(x,t)$ is governed by the Fokker-Planck partial differential equation (Risken, 1996):

$$p_t(x, t) = [d\, p_x(x, t)]_x - \{f(x, \bar{u})\, p(x, t)\}_x, \; 0 \le x < \infty, \; t > 0 \qquad (2a)$$

$$x = 0: \; d\, p_x(0, t) - f(0, \bar{u})\, p(0, t) = 0, \quad x = \infty: \; p_x(\infty, t) = 0 \qquad (2b\text{-}c)$$

$$t = 0: \; p(x, 0) = p_0(x), \qquad d = q^2/2 \qquad (2d)$$

where d is the "diffusion constant" set by the noise intensity, (2b)-(2c) is the boundary condition pair and (2d) is the initial condition with initial PDF p_o. Condition (2b) establishes that x can have only positive values (Gardiner, 1997), in the understanding that this condition is easily met by writing the chemical process states in suitable scales.

2.1 Stationary probability density function

The stationary solution of (2) is given by:

$$p_s(x) = N_0 e^{-\frac{\phi(x)}{d}}, \quad \phi(x) = -\int_x f(s)ds \qquad (3a\text{-}b)$$

where N_0 is the integration constant associated to the normalization of $p_s(x)$ and $\phi(x)$ is the potential function.

From the examination of the stationary solution (3) in the light of multiplicity features of the deterministic system, the next conclusions follow. When the deterministic system has a unique global attractor $\bar{x} \in X$, the potential function $\phi(x)$ has a single well shape with minimum at $\bar{x}$, and the stationary PDF $p_s(x)$ is monomodal with maximum at $\bar{x}$, meaning that the solution $\bar{x}$ is the more probable state over X. As noise intensity decreases (d tends to zero) the monomodal PDF tends to the Dirac Delta function $\delta(x - \bar{x})$ about $\bar{x}$. When the deterministic system has multiple steady state $\bar{x}_1, \ldots, \bar{x}_m \in X$,

with domains of attraction $X_1,\ldots, X_m$ such as $\bigcup_{i=1}^{m} X_i = X$: (i) the potential function $\phi(x)$ has a multi well shape potential with minima at $\bar{x}_1,\ldots, \bar{x}_m$, (ii) the multivalued stationary PDF $p_s(x)$ has maxima at $\bar{x}_1,\ldots, \bar{x}_m$, (iii) the most probable steady state solution is the one with the deepest potential well $\phi(\bar{x}_m)$ and therefore with the largest maximum, and (iv) the difference among PDF maxima grows exponentially with the decrease of d. As a consequence of (iv), at low d values the distribution appears monomodal and tends to a Dirac Delta when the noise intensity tends to zero. Multimodality is maintained, even at low d values, when the potential minima are equal and in this case the limit as d tends to zero is a multi Dirac Delta.

2.3 *Probability distribution function evolution*

The right hand side of (2a) can be written as follows:

$$p_t = d\, p_{xx} - f(x, \bar{u})\;\; p_x - f_x(x, \bar{u})\, p \tag{4}$$

evidencing that: (i) the shape of the PDF over time is due to a source/sink mechanism $-f_x\, p$ combined with two transport mechanisms, one diffusive $d\, p_{xx}$ and one convective $-f\, p_x$, and (ii) the PDF temporal evolution is obtained by giving an initial value $p(x,0) = p_0(x)$ and integrating numerically the FP equation. If the deterministic system has a unique global attractor, the potential function has a single minimum, and the PDF reaches asymptotically a monomodal distribution, regardless the initial PDF shape. Otherwise, when there is deterministic steady-state multiplicity with multiple potential minima, the PDF evolution may exhibit some behaviors, which seem atypical from a deterministic nonlinear system perspective. In fact, the PDF settles at some multimodal PDF with largest maximum at (probability around) the attractor $\bar{x}_1$, then after some time, the PDF eventually starts moving and reaches another multimodal shape with a different largest maximum at (probability around) the attractor $\bar{x}_2$. In fact, for the case of steady-state multiplicity with an asymptotic (stationary) bimodal PDF, the time necessary for a state x at the steady state $x = \bar{x}_1$, with domain of attraction X_1, to escape to the steady-state $x=\bar{x}_2$, with domain of attraction X_2, is approximated by the formula (Gardiner, 1997):

$$T \propto \exp[(\phi(\bar{x}_2)-\phi(\bar{x}_1))/d] \tag{5}$$

which resembles Arrhenius' equation in chemical kinetics.

Thus stationary-to-stationary ($\bar{x}_1$-to-$\bar{x}_2$) state transition probability is favored by: i) a small well potential difference $[\phi(\bar{x}_1)-\phi\bar{x}_2)]$ and (ii) a well potential with large minima. When the minima have the same ordinate, there is not a dominant attractor and the probability of leaving one of the wells is the same.

3. STOCHASTIC MODEL OF AN ISOTHERMAL CSTR

3.1 CSTR with Langmuir-Hinshelwood kinetics

As a representative example in catalytic reactors, let us consider an isothermal CSTR with Langmuir – Hinshelwood kinetics, with the corresponding mass balance being described by the nonlinear differential deterministic system:

$$\dot{x} = f(\mathrm{x}, Da, \sigma),\, x(0) = x_o, \tag{6}$$
$$f(x, Da, \sigma) = (1 - x) - Da(1 + \sigma)^2\, x/(1 + \sigma x)^2$$
$$x = c/c_i,\;\; \tau = t/(V_R/Q),\;\; Da = (V_R/Q)k/(1 + \sigma)^2,\; \sigma = K\, c_i.$$

x is the dimensionless concentration (referred to the feed concentration c_i), t and τ are, respectively, the actual and dimensionless time, Q the volumetric feedrate, V_R the reactor volume, k the reaction-rate constant, K the equilibrium adsorption constant and Da the Damkohler number. In spite of its simplicity, the above single-state system exhibits a rather rich behavior over the parameter space pair (Da, σ), showing multiple steady-states for a specified range of parameter values. In the case of multiplicity, there are two (low and high concentration) stable steady-states and one (intermediate concentration) unstable steady-state. Moreover system (6) captures the important nonlinearities which underline the lack of global and local observability at the value $x = 1/\sigma$ (where the reaction rate is maximum), in the understanding that this feature makes difficult the design of nonlinear observers and controllers of an important class of chemical reactors with nonmonotonic kinetics (Schaum *et al*., 2008).

The stochastic system associated to the deterministic reactor (6) is given by (1) replaced by $f(x,Da,\sigma)$, and the corresponding stationary PDF is given by:

$$p_s(x) = N_0 \exp\left[-\frac{1}{d}\left(-x + \frac{x^2}{2} + \frac{Da(1+\sigma)^2}{\sigma^2(1+\sigma x)} + \frac{Da(1+\sigma)^2 \ln(1+\sigma x)}{\sigma^2}\right)\right]. \tag{7}$$

3.1 *Deterministic nonlinear dynamics*

The bifurcation analysis of system (6) evidences the occurrence of saddle-node bifurcation when $Da > 0$ (see Figure 1) and, on the parameter space (Da, σ), the deterministic reactor steady-state (SS) exhibits either: (i) a unique global attractor $\bar{x}$ with domain of attraction X[0, 1], or (ii) three-SS multiplicity, with two (low and high concentration) stable and one (intermediate concentration) unstable steady-state.

In the multiplicity case, there are two basins of attraction (X_1 and X_2), one per attractor. Thus, in the single SS case any state motion $x(t)$ beginning in $x_0 \in X$ remains in X, and asymptotically converges to the steady state $\bar{x}$ in X (see Figure 2a):

$$x_o \in X = [0, 1] \;\Rightarrow\; x(t) \in X,\; x(t) \to \bar{x}$$

In the three-SS case (with two stable attractors $\bar{x}_i$, i = 1, 2 with domain of attraction X_i) any state motion $x(t)$ beginning in $x_o \in X_i$ remains in X_i, and asymptotically converges to the steady-state $\bar{x}_i$ in X_i, this is (see Figure 2b):

$$x_o \in X_i = [0, 1] \Rightarrow x(t) \in X_i,\ x(t) \to \bar{x}_i,\ i = 1, 2$$

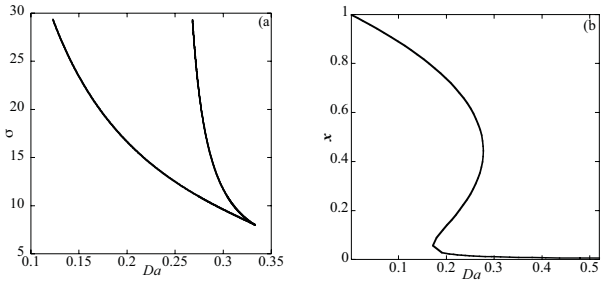


Figure 1: a) bifurcation diagram of system (6) and b) corresponding solution diagram at σ=20.

In particular, for σ = 20, the deterministic reactor system (6) exhibits: (i) a low (or high) concentration unique global attractor for $0 < Da < Da^- \approx 0.172$ (or $Da > Da^+ \approx 0.277$), (ii) three steady-states for $Da^- < Da < Da^+$, and (iii) two saddle-node bifurcations at Da equal to Da^- and Da^+ (see Figure 1).

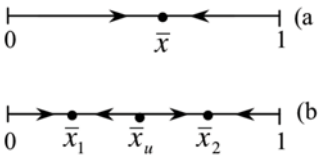


Figure 2: Phase diagram (a) in the single-SS case and (b) in the three-SS case.

3.2 *Stationary stochastic behavior*

The stationary (asymptotic) behavior of the PDF which satisfies the FP equation was investigated by setting σ equal to 20 (cf. Section 3.1), varying the value of Damkohler number $0 < Da < 1.0$ and the noise-related diffusion coefficient $10^{-5} < d < 10^{-3}$ (Ratto, 1998). The normalization constant in (3a) was calculated through the orthogonal collocation method on finite elements.

In Figure 3a (or 3b) the stationary PDF for $Da = 0.226$ (or $Da = 0.231$) with three SSs and two attractors, for two noise levels $d = 5.0\ 10^{-4}$ (continuous line) and $5.0\ 10^{-3}$ (dashed line) is shown. At the lowest d value only one peak is clearly detectable at $x \approx 0.683$ (or $x \approx 0.0178$), while the second peak corresponding to $x \approx 0.018$ (or $x \approx 0.671$) becomes evident only at the highest d value.

In Figure 4a (or 4b) is presented the potential function $\phi(x)$ (or stationary PDF for $d = 10^{-4}$) at three values of Da: 0.226 (dotted line), 0.229 (continuous line), and 0.231 (dashed line). In accordance with the deterministic bistability properties there are two attracting minima for the potential $\phi(x)$, meaning the possibility of well-to-well steady-state transition with longer residence in the deepest well. As expected, at low diffusion value only one peak is clearly visible for $Da = 0.226$ (extinction) and for $Da = 0.231$ (ignition). When the two minima have the same value, $Da \approx 0.229$, the stationary PDF exhibits bimodality made of nearly non overlapping monomodal PDFs or equivalently, a well-to-well potential without a dominant attractor.

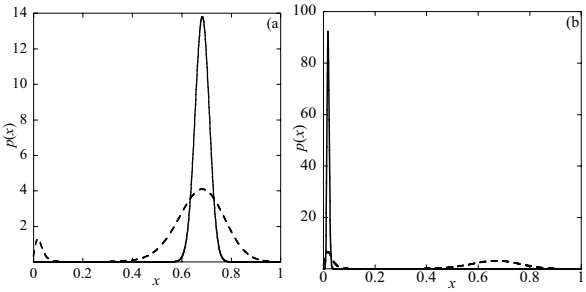


Figure 3: Stationary PDF when a) $Da = 0.226$ and b) $Da = 0.231$ for $d = 5.0\ 10^{-4}$ (solid line) and $d = 5.0\ 10^{-3}$ (dashed line).

The latter case could be considered as an important bifurcation characteristic related to the stochastic behavior, and not to the deterministic one. This Damkohler critical number Da_C is determined by the enforcement of the next equipotential conditions:

$$\left.\frac{d\phi}{dx}\right|_{(\bar{x}_1;Da_C)} = \left.\frac{d\phi}{dx}\right|_{(\bar{x}_2;Da_C)} = 0 \qquad (\bar{x}_1 \neq \bar{x}_2)$$
$$\phi(\bar{x}_1;Da_c) = \phi(\bar{x}_2;Da_c) \qquad (8)$$

In conclusion, the Da_C value corresponds to a transition between two qualitatively different behaviors of the stochastic reactor system. This transition appears smooth for high d values, meaning that a bimodal distribution is apparent in a wider neighborhood of Da_C, and becomes sharper as the diffusion coefficient tends to zero.

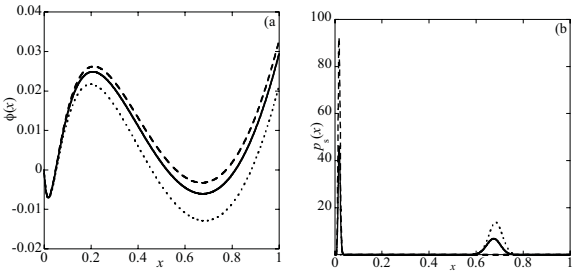


Figure 4: a) Potential function and b) stationary PDF (d=10^{-4}) for different Da values: Da=0.226 (dotted line), Da=0.229 (solid line), Da=0.231 (dashed line).

The one-dimensional manifold satisfying (8) can be derived by resorting to standard continuation algorithms (Doedel *et*

al., 1997), and the stochastic bifurcation diagram, over the (*Da*-σ) plane, was constructed and reported in Figure (5) together with the bifurcation diagram of (6).

Observe that the passage from the deterministic (Figure 1a) to the stochastic (Figure 5) bifurcation diagram evidences: (i) the correspondence between the deterministic steady-state and stochastic stationary nonlinear features, and (ii) the kind of information contained in the stochastic diagram and not in the deterministic one.

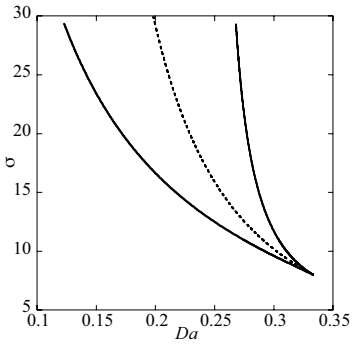


Figure 5: Diagram of the saddle-node bifurcation of the deterministic system (solid line) and the (Da_C, σ) curve (dashed line).

3.3 *Dynamic behavior*

According to the preceding developments, in a deterministic framework, the domain of attraction determines the steady-state which will be reached asymptotically by the system. However, from a stochastic point of view it may happen that one of the deterministic steady states has a low or negligible asymptotic probability of being reached, regardless the initial condition.

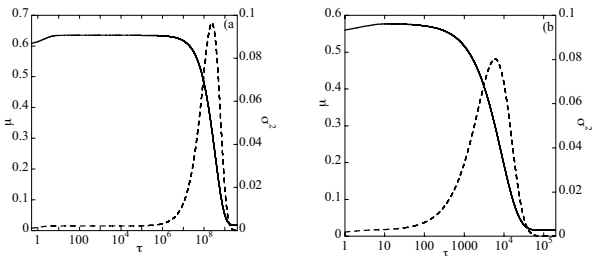


Figure 6: Dynamic behavior of mean (solid line) and variance (dashed line) of the PDF when $d = 10^{-3}$ at a) Da = 0.244 and b) Da = 0.260. The time scale is logarithmic.

Figure 6 represents the transient of mean and variance of the probability distribution function when $d = 10^{-3}$, and the initial condition is a Gaussian distribution with mean equal to 0.6 and variance equal to 0.02, at Da = 0.244 (Figure 6a) and Da = 0.260 (Figure 6b). In both cases, the absolute minimum of the potential function is positioned on the lower branch of the solution diagram, but the initial distribution is inside of the basin of attraction of the other solution, meaning that the probability that the initial condition is outside the weaker attractor is almost negligible.

The responses of the PDF show that during the transient, the mean of the distribution does not directly move towards its steady state value in the ignited zone, but first approaches the higher solution. It should be noted that, at Da = 0.244 (Figure 6a), mean and variance are almost constant for a wide interval of time (the time scale in Figure 6 is logarithmic), looking as if a stable stationary solution was definitely reached. Thus, the high concentration solution appears as a *quasi-stationary* solution. In other words, only after a long transient the system departs from the extinction steady-state and eventually reaches the ignited region. The variance reaches a maximum during the transition from the quasi-stationary to the stationary solution, implying that the PDF becomes bimodal with its two peaks corresponding to the two deterministic attractors. As time elapses, one of the peaks becomes negligible and the other one finally prevails. When Da = 0.260 (Figure 6b), the system again moves first towards the solution contained in the attraction basin where the initial distribution is centered (low conversion solution), but after a while the mean starts decreasing towards its stationary value. Some snapshots of the evolving probability distribution are shown in Figure 7 for Da = 0.244. It must be pointed out that the *quasi-stationary* condition duration can range from several to orders of magnitude the reactor natural deterministic dynamics (set by the residence time), depending on the noise intensity, and this is a fact that must be carefully accounted for in long-term prediction assessments, with applicability in safe process design.

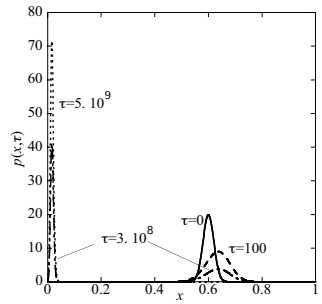


Figure 7: Snapshots of the PDF at τ=0 (solid line), τ =100 (dashed line), τ =3.0 10^8 (dashed-dotted line) and τ =5.0 10^9 (dotted line).

The duration of the *quasi-stationary* state can be related to the escape time, evaluated by means of (5). Calculating the escape time for Da = 0.244 and Da = 0.260 we found, respectively, T_1=4.3 10^8 and T_2=8.4 10^3. These results establish that stationary conditions are reached for a time greater than the calculated escape time, as confirmed by the simulation. The decreasing of the escape time as Da approaches the bifurcation value reflects the fact that the relative minimum is less and less deep until it disappears at the bifurcation point.

6. CONCLUSIONS

The global-nonlinear stochastic behavior of the concentration in an isothermal CSTR reactor with multistability has been characterized on the basis of standard deterministic tools in conjunction with FP equation theory. In addition to issues considered in previous studies in chemical reactor (Pell and Aris, 1969; Ratto 1998) and combustion engineering (Oberlack, 2000), in this study the presence of multistability, transient behavior, and the connection between deterministic and stochastic modeling approaches were considered. In particular, the interplay between the stochastic (mono or multimodality, potential, quasi-stability, and escape time) and deterministic (stability, multiplicity and bifurcation) features was identified. The stationary analysis revealed that, even when multistability was expected for the deterministic model, the probability distribution function usually appeared as monomodal, indicating that there is one dominant attractor, with higher probability of being reached asymptotically. However, the occurrence of multi-stabilities in the deterministic model did affect the behavior of the transient dynamics and the system could stay in a neighborhood of the weaker attractor for a long time interval, thus appearing as a *quasi-stationary* state.

The results of this paper constitute a point of departure: (i) to study the multi-state nonlinear stochastic system case, and (ii) to explore the implications and applications for global nonlinear estimation, control, and safe process designs.

*A*cknowledgement.

J. Alvarez kindly acknowledges Regione Sardegna for the support, through the program "Visiting Professor 2008", for the realization of this work at the Dipartimento di Ingegneria Chimica e Materiali of the University of Cagliari.

REFERENCES

Doedel, E. J., Champneys, A. R., Fairgrieve, T. F., Kuznetsov, Y. A., Sanstede, B., and Wang, X., (1997). "AUTO97: continuation and bifurcation software for ordinary differential equations".

Dutta, S., Chowdhury, R., and Bhattacharya, P., (2001). Parametric sensitivity in bioreactor: an analysis with reference to phenol degradation system. *Chem. Eng. Sci*., 56, 5103-5110.

Gardiner, C. W., (1997). *Handbook of stochastic methods*. Springer-Verlag, Germany.

Hanggi, P. and Jung, P., (1988). Bistability in active circuits: Application of a novel Fokker-Planck approach. *IBM J. Res. Develop*., 32(1), 119-126.

Hauptmanns, U., (2008). Comparative assessment of the dynamic behaviour of exothermal chemical reaction including data uncertainties. *Chem. Eng. J*., 140, 278-286.

Horenko, I., Lorenz, S., Schutte, C., and Huisinga, W., (2005). Adaptive approach for nonlinear sensitivity analysis of reaction kinetics. *J. Comp. Chem.*, 26(9), 941-948.

Huang, D. W., Wang, H. L., Feng, J.F. , and Zhu, Z.W., (2008). Modelling algal densities in harmful algal blooms (HAB) with a stochastic dynamics. *Applied Mathematical Modelling,* 32(7), 1318-1326.

Lo, C. F., (2007). Stochastic Gompertz model of tumor cell growth. *Journal of Theoretical Biology* 248, 317-321.

Mei, D. C., Xie , C.W. and Zhang, L., (2004). The stationary properties and the state transition of the tumor cell growth model. *European Physical Journal B* 41(1) 107-112.

Morbidelli, M., and Varma, A., (1989). A generalized criterion for parametric sensitivity: Application to a pseudohomogeneous tubular reactor with concecutive or parallel reactions. Chem. Eng. Sci., 44, 1675-1696.

Oberlack, M., Arlitt, R., and Peters, N., (2000). On stochastic Damkohler number variations in a homogeneous flow reactor. *Combust. Theory Modelling*, 4, 495-509.

Paladino, O., and Ratto, M., (2000). Robust stability and sensitivity of real controlled CSTRs. *Chem. Eng. Sci*., 55, 321-330.

Pell, T. M., and Aris, R., (1969). Some problems in chemical reactor analysis with stochastic features. *I&EC Fundamentals*, 8(2), 339-345.

Rao, N. J., Ramkrishna, D., and Borwanker, J. D., (1974). Nonlinear stochastic simulations of stirred tank reactors. *Chem. Eng. Sci*., 29, 1193-1204.

Ratto, M., (1998). A theoretical approach to the analysis of PI-controlled CSTRs with noise. *Comp. Chem. Eng.*, 22(11), 1581-1593.

Ratto, M., and Paladino, O., (2000). Analysis of controlled CSTR models with fluctuating parameters and uncertain parameters. *Chem. Eng. Sci*., 79, 13-21.

Risken, H., (1996). *The Fokker-Planck equation: Methods of solutions and Applications*. Springer-Verlag, Berlin.

Schaum A, Moreno J. A., Díaz-Salgado, J., and Alvarez J. (2008). Dissipativity-based observer and feedback control design for a class of chemical reactors. *Journal of Process Control*, 18(9): 896–905

Sherer E., Ramkrishna, D., (2008). Stochastic analysis of multistate systems. *Ind. Chem. Eng. Res*., 47(10), 3430-3437.

Soboleva, T. K., and Pleasants, A.B., (2003). Population growth as a nonlinear stochastic process. *Mathematical and Computer Modelling,* 38(11-13), 1437-1442.

Wiggins, S., (1990). *Introduction to applied nonlinear dynamical systems and chaos*, Springer-Verlag, New York.

Process Control and Optimization

Poster Session

Nonlinear Model Predictive Control Using Multiple Shooting Combined with Collocation on Finite Elements

Jasem Tamimi and Pu Li

Simulation and Optimal Processes Group, Institute of Automation and Systems Engineering, Ilmenau University of Technology, P. O. Box 10 05 65, 98684 Ilmenau, Germany. (Tel.:+49-3677-691427, e-mail:{jasem.tamimi/pu.li}@tu-ilmenau.de)

Abstract: A new approach to nonlinear model predictive control (NMPC) is proposed in this paper. The multiple shooting method is used for discretizing the dynamic system, through which the optimal control problem is transformed to a nonlinear program (NLP). To solve this NLP problem state variables and their gradients at the end of each shooting need to be computed. Here we employ the method of collocation on finite elements to carry out this task. Due to its high numerical accuracy, the computation efficiency for the integration of model equations can be enhanced, in comparison to the existing multiple shooting method where an ODE solver is applied for the integration and the chain-rule for the gradient computation. The numerical solution framework is implemented in C++. Two examples are taken to demonstrate the effectiveness of the proposed NMPC algorithm.

Keywords: Optimal control, NMPC, multiple shooting, collocation on finite elements.

1. INTRODUCTION

Solving optimal control problem is highly motivated nowadays, since these solutions are very important in almost all industrial fields such as chemical, electrical, mechanical, and economical systems. One of the optimal control algorithms is MPC which refers to a class of computer control strategies that utilize an explicit process model to predict the future response of the plant (Qin and Badgwell, 2003). MPC, also known as receding horizon control, has the ability to handle input as well as output constraints and transparent tuning capabilities (Gatlu and Zafiriou, 1992).

The main goal of MPC is to find an optimal vector of control functions that minimize or maximize a performance index subject to a given process model (usually a nonlinear differential equation system) as equality constraints, and boundary conditions as inequality constraints on the states and controls. Simple problems can be solved by the so-called *indirect method* which is based on the first order optimality condition of variation (Diehl *et al.,* 2006, Schäfer *et al.*, 2007). This leads to a two-point-boundary value problem in ordinary differential equations (ODE). For more details see e.g. Bryson and Ho (1975), Kirk (1970), and Lewis and Syrmos (1995).

On the other hand, the *direct method* which follows the philosophy of "first discretize then optimize", transforms the optimal control problem into a NLP problem which can then be solved by the method of sequential quadratic programming (SQP). In this way inequality constraints and equality path constraints can be easily treated, and we can also successfully deal with highly nonlinear complex optimal control problems (Diehl *et al.*, 2006, Schäfer *et al.*, 2007).

In all direct methods, the control trajectory will be parameterized and the state trajectories computed using either sequential or simultaneous approaches. In the *sequential approach*, the state variables are considered as an implicit function of control trajectories, where the ODEs are addressed as an initial value problem using one of the dedicated integration methods like Runge-Kutta or Euler algorithms (Sargent and Sullivan, 1977; Kraft, 1985; Biegler *et al.,* 2002). In the *simultaneous approach*, state trajectories are parameterized, too, and we deal with all of parameterized variables (states and controls) as optimization variables in the NLP. The ODEs will be represented as equality constraints, either with collocation on finite elements (Biegler *et al.,* 2002; Hong *et al.,* 2006; Li, 2007) or with multiple shooting (Bock and Plitt, 1984; Leineweber, 1995; Diehl, 2001; Diehl *et al.,* 2002a; Diehl *et al*., 2002b).

In this work, we propose a new approach to the solution of nonlinear model predictive control (NMPC) problems. This control strategy is a combination of the multiple shooting and the collocation method. We use multiple shooting for discretizing the dynamic system, so that the optimal control problem is transformed to a NLP problem in which continuity conditions in each shooting are considered as equalities and state constraints at the end of each shooting as inequalities. To solve this NLP problem the values of state variables and their gradients at the end of each shooting have to be computed. Here we employ collocation on finite elements to carry out this task. Due to its high numerical accuracy, the computation efficiency for the integration of the ODEs can be enhanced, in comparison to the existing multiple shooting method where an ODE solver is applied for the integration and chain-rules for the gradient computation. We implement the proposed approach with a numerical solution framework in C++. Two examples are taken to demonstrate the effectiveness of the proposed NMPC algorithm. The results from our approach are compared with

those achieved from the multiple shooting method (using the software MUSCOT II (Diehl *et al.*, 2001)).

2. NONLINEAR MODEL PREDICTIVE CONTROL

2.1 Optimal control problem

We will consider the following optimal control problem

$$\min \int_{t_0}^{t_f} L(x(t),u(t),t)dt + E\left(x(t_f)\right)$$

s.t.

$$\begin{aligned} &\text{(i)} && x(t_0) = x(0), \\ &\text{(ii)} && \dot{x}(t) = f(x(t),u(t),t),\ t \in [t_0,t_f] \\ &\text{(iii)} && g(x(t),u(t),t) \geq 0\ ,\ t \in [t_0,t_f] \\ &\text{(iv)} && r\left(x(t_f)\right) = 0, \end{aligned} \tag{1}$$

where $x(t), u(t)$ are the state and control variables, respectively, t_0 and t_f are initial and final time of the receding horizon, and constraint (i) is the initial value condition, (ii) the nonlinear ODE model, (iii) the path constraint, and (iv) the terminal constraint.

2.2 Direct multiple shooting scheme

The direct multiple shooting algorithm proposed by Bock and Plitt (1984) for solving problem (1) can be summarized in the following steps:

1) Discretize the time horizon $[t_0, t_f]$ into equal subintervals $[t_i, t_{i+1}]$, such that

$$t_0 < t_1 < \cdots < t_n = t_f \tag{2}$$

where *n* is the total number of subintervals.

2) Parameterize the control function $u(t)$ for each subinterval:

$$u(t) = v_i \quad for \quad t \in [t_i, t_{i+1}] \tag{3}$$
$$i = 0,1,\dots,n-1$$

3) Parameterize the initial condition of the state vector for each subinterval:

$$x(t_i) = h_i \tag{4}$$
$$i = 0,1,\dots,n-1$$

4) Evaluate the state trajectories in each subintervals and the value of h_i from the final state subinterval considering the parameterized state initial value in the previous step:

$$\dot{x}_i(t) = f(x_i(t),v_i,t),\ t \in [t_i, t_{i+1}] \tag{5a}$$
$$x_i(t_i) = h_i \tag{5b}$$

5) Define the continuity constraints:

$$h_{i+1} - x_i(t_{i+1}; h_i, v_i) = 0 \tag{6}$$

6) Compute the objective function for each subinterval, so we need to solve the following NLP

$$\min_{h_i,v_i} \sum_{i=0}^{n-1} \int_{t_i}^{t_{i+1}} L(x_i(t),v_i)dt + E\left(x_n(t_n)\right)$$

s.t. (7)

$$h_0 - x(0) = 0$$
$$h_{i+1} - x_i(t_{i+1}; h_i, v_i) = 0, i = 0,1,\dots,n-1,$$
$$g(h_i, v_i) \geq 0$$

Eq. (7) can be described as

$$\min_w\ A(w) \qquad s.t. \qquad \begin{cases} B(w) = 0 \\ C(w) \geq 0 \end{cases} \tag{8}$$

where $w = [h_0, v_0, h_1, v_1, \dots, h_n, v_{n-1}]$,

$$B(w) = \begin{bmatrix} h_0 - x(0) \\ h_1 - x_0(t_1; h_0, v_0) \\ \vdots \\ h_{n-1} - x_{n-2}(t_{n-1}; h_{n-2}, v_{n-2}) \end{bmatrix},$$

$$C(w) = \begin{bmatrix} g(h_0, v_0) \\ g(h_1, v_1) \\ \vdots \\ g(h_{n-1}, v_{n-1}) \end{bmatrix}.$$

We can use the spars nonlinear optimizer (SNOPT) to solve the above NLP problem. In SNOPT equality constrains will be transformed into inequality constraints by introducing a set of slack variables, i.e.

$$\min_w A(w)$$

s.t. (9)

$$\left(\begin{pmatrix} B(w) \\ C(w) \end{pmatrix} - s = 0\right), \text{ and } \ l \leq \begin{pmatrix} w \\ s \end{pmatrix} \leq u$$

where $s = (s_0, \cdots, s_{n-1}, s_n, \cdots s_{2n-2})^T$. For more information on SNOPT see Gill *et al.* (2005) and Gill *et al.* (2008). Consequently, problem (8) can now be rewritten as:

$$\min_w A(w)$$

s.t. (10)

$$l \leq D(w) \leq u$$

$$\text{where, } D(w) = \begin{bmatrix} h_0 - x(0) \\ h_1 - x_0(t_1; h_0, v_0) \\ \vdots \\ h_{n-1} - x_{n-2}(t_{n-1}; h_{n-2}, v_{n-2}) \\ g(h_0, v_0) \\ g(h_1, v_1) \\ \vdots \\ g(h_{n-1}, v_{n-1}) \end{bmatrix},$$

$$l = \begin{bmatrix} l_0 \\ \vdots \\ l_{n-1} \\ l_n \\ \vdots \\ l_{2n-2} \end{bmatrix} = \begin{bmatrix} 0 \\ \vdots \\ 0 \\ 0 \\ \vdots \\ 0 \end{bmatrix} \ and \ u = \begin{bmatrix} u_0 \\ \vdots \\ u_{n-1} \\ u_n \\ \vdots \\ u_{2n-2} \end{bmatrix} = \begin{bmatrix} 0 \\ \vdots \\ 0 \\ \infty \\ \vdots \\ \infty \end{bmatrix}.$$

2.3 SQP iteration

Fig. 1 shows all of the information needed for each SQP iteration.

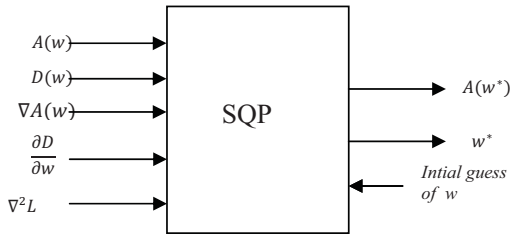


Fig. 1: Inputs and outputs of each SQP iteration.

In Fig. 1, $\nabla A(w)$ and $\frac{\partial D}{\partial w}$ are the gradient of the objective function and Jacobian of the equality constraints in (7), respectively. $A(w^*)$ and w^* are the objective function value and the optimization variables at the solution. ∇^2L denotes the Hessian of the Lagrangian. The sensitivity information, i.e. $\nabla A(w)$ and $\frac{\partial D}{\partial w}$, plays the most important role in the SQP iteration and its computation requires much CPU-time. In the existing multiple shooting algorithm it is done by integrating the ODEs with an ODE solver and then using the chain-rule for the sensitivity computation. In this work we employ the method of collocation on finite elements to carry out the ODE integration and compute these sensitivities for each shooting. This proposed method is described in the next section.

3. SOLVING ODE AND SENSITIVITIES

To solve NLP (9) we have to solve the set of ODEs (5a). If we use piece-wise constant parameters for v_i, we can rewrite the ODE in each subinterval as

$$\begin{pmatrix}\dot{x}_i(t)\\ \dot{v}_i\end{pmatrix} = \begin{pmatrix}f(x_i(t), v_i, t)\\ 0\end{pmatrix} \Rightarrow \begin{matrix}\dot{z}_i = f(z_i(t), t)\\ z_i(t_i) = [x_i\ v_i]^T\end{matrix} \quad (11)$$

Using collocation method the state variables $z_i(t)$ will be approximated by the following Lagrangian polynomials (Finlayson, 1980)

$$\mathrm{z}(t) = \sum_{j=0}^{M}\left[\prod_{\substack{k=0\\ j\neq k}}^{M}\frac{(t-t_k)}{(t_j-t_k)}\right] z_j \quad (12)$$

where M is the number of the collocation points. Using the three-point-collocation to compute the vector z, we yield

$$z(t) = T_0 z_0 + T_1 z_1 + T_2 z_2 + T_3 z_3 \quad (13)$$

where $T_j = \prod_{\substack{k=0\\ k\neq j}}^{3}\frac{t-t_k}{t_j-t_k}$.

To define the end time point of a subinterval to be the beginning point of the next one, we yield inside each shooting

$$t_0 = t_i, \quad t_1 = \alpha_1(t_{i+1} - t_i),$$
$$t_2 = \alpha_2(t_{i+1} - t_i), \text{ and } t_3 = t_{i+1} \quad (14)$$

where $\alpha_1 = 0.127$ and $\alpha_2 = 0.5635$. Then

$$\dot{T}_{i,k} Z_{i,k} + \dot{T}_{i,0} Z_{i,0} - f_{i,k}(z_i(t), t) = 0 \quad (15)$$

where $T_{i,k} = \begin{bmatrix} T_{i,1}(t_1) & T_{i,2}(t_1) & T_{i,3}(t_1)\\ T_{i,1}(t_2) & T_{i,2}(t_2) & T_{i,3}(t_2)\\ T_{i,1}(t_3) & T_{i,2}(t_3) & T_{i,3}(t_3)\end{bmatrix}, Z_{i,k} = \begin{bmatrix} z_{i,1}\\ z_{i,2}\\ z_{i,3}\end{bmatrix}$

$$T_{i,0} = \begin{bmatrix} T_{i,1}(t_0) & 0 & 0\\ 0 & T_{i,2}(t_0) & 0\\ 0 & 0 & T_{i,3}(t_0)\end{bmatrix}, Z_{i,0} = \begin{bmatrix} z_{i,0}\\ z_{i,0}\\ z_{i,0}\end{bmatrix}$$

We solve the nonlinear equations (15) on the collocation points by using the Newton-Raphson method to find $Z_{i,k}$ and z_i. The first Taylor-expansion of (15) leads to

$$\dot{T}_{i,k}\frac{\partial z_{i,k}}{\partial z_{i,0}} + \dot{T}_{i,0} - \frac{\partial f_{i,k}(z_i(t),t)}{\partial z_{i,0}} = 0 \quad (16)$$

We define $\frac{\partial z_{i,k}}{\partial z_{i,0}} = \Psi_{i,k}$, then

$$\dot{T}_{i,k}\Psi_{i,k} + \dot{T}_{i,0} - \frac{\partial f_{i,k}(z_i(t),t)}{\partial z_{i,k}}\Psi_{i,k} = 0 \quad (17)$$

or

$$\Psi_{i,k} = -\left[\dot{T}_{i,k} - \frac{\partial f_{i,k}(z_i(t),t)}{\partial z_{i,k}}\right]^{-1}\dot{T}_{i,0} \quad (18)$$

In fact, equation (18) is a linear equation system and thus can be solved by a LU factorization using forward and backward substitution, for more details see Golub and van Loan (1996). From the computed value of $\Psi_{i,3}$ we receive the Jacobian $\frac{\partial D}{\partial w}$, since

$$\frac{\partial D}{\partial w} = \begin{bmatrix} I & 0 & 0 & 0 & 0 & 0 & 0 & 0 & \cdots & 0\\ \Psi_{0,3} & I & 0 & 0 & 0 & 0 & 0 & \cdots & & 0\\ 0 & 0 & \Psi_{1,3} & I & 0 & 0 & 0 & \cdots & & 0\\ 0 & 0 & 0 & 0 & \Psi_{2,3} & I & 0 & \cdots & & 0\\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \ddots & 0\\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & \cdots & \Psi_{n-1,3} & I\end{bmatrix} \quad (19)$$

where I is a unit matrix. In the same way, we can calculate the gradient vector of $\nabla A(w)$.

4. THE PROPOSED ALGORITHM

As we have seen above, the multiple shooting method depends mainly on the SQP iteration. Inside each SQP iteration the gradient values of the objective function and Jacobian of the constraints as well as the approximated Hessian need to be computed. Based on the theoretical development in Sections 2 and 3, we propose the following algorithm to solve the nonlinear optimal control problem.

Algorithm 1:

1. *Initialize SQP*
   - 1.1. *Time horizon.*
   - 1.2. *Subintervals.*
   - 1.3. *Upper and lower bounds for states, controls and constraints.*
   - 1.4. *Fixed initial value constraint.*
   - 1.5. *Initial guess.*

2. *Define the continuity constraints $B(w)$ (8).*
3. *Define the continuity constraints $C(w)$ (8).*
4. *Initialize the three collocation points for each subinterval (14).*
5. *Compute the constraint equations and their sensitivities*
   5.1. *Define collocation equations (15).*
   5.2. *Solve (15) using Newton-Raphson.*
   5.3. *Define sensitivity equations (1).*
   5.4. *Solve (17) using LU factorization.*
6. *Compute objective function and its sensitivity.*
7. *Solve SQP iteration*
   7.1. *If KKT is not satisfied go to 4.*
8. *End*

This algorithm is realized in the framework of the numerical algorithm group (NAG) library Mark 8 (Numerical Algorithms Group Ltd, 2005) and IPOPT (Wächter, 2008) for SQP and in C/C++ for the rest of computations.

5. A CASE STUDEIS

We consider the following optimal control problems to demonstrate the performance of the proposed algorithm.

Example 1: Batch reactor - temperature profile. Maximize yield of x_2 after one hour's operation by manipulating a transformed temperature $u(t)$. This example is taken from Diehl *et al.* (2001).

$$\max_{u,x1,x2} \quad x_2(t_f)$$

s.t. (20)

$$\dot{x}_1(t) = -(u(t) + \frac{u^2(t)}{2})x_1(t)$$
$$\dot{x}_2(t) = u(t)x_1(t), \qquad t \in [0,1]$$
$$x_1(0) = 1, x_2(0) = 0.$$
$$0 \le x_1(t), x_2(t) \le 1$$
$$0 \le u(t) \le 5$$

We discretize the dynamic system with 20 subintervals. The computation was done using a PC with an intel processor "Pentium 4", 3 GHz and 1G Byte RAM. The solution took 350 ms and provides the final value of objective function with $x_2(t_f) = 0.57329$. Fig. 2 shows the optimal control trajectory and Fig. 3 the corresponding state trajectory x_1 while x_2 is shown in Fig. 4. These profiles of states (x_1 and x_2) and optimal control trajectory are identical, by using both MSCOD II and the proposed algorithm.

If we solve this problem with different number of subintervals, e.g. 5, 10, 20, 40, 80 and 160 subintervals, we can note from the results, as shown in Table 1, that the number of optimization variables (z) and the number of constrains will be increased when the number of subintervals increases. The CPU-time will increase exponentially. However, if we compare the CPU-time taken by MUSCOD II with that of the proposed algorithm, it can be seen at a large number of subintervals (i.e. a high dimension of the NMPC), the proposed algorithm will be more effective.

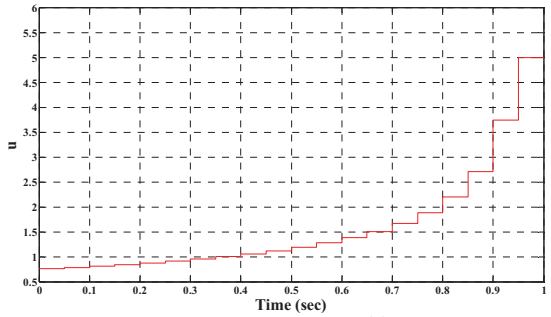


Fig. 2: The optimal control trajectory $u(t)$

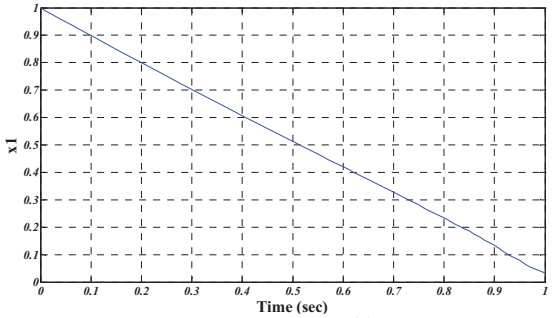


Fig. 3: The optimal state trajectory $x_1(t)$

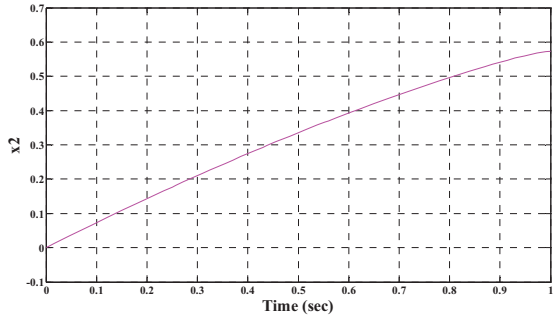


Fig. 4: The optimal state trajectory $x_2(t)$

Table 1: Results of using different number of subintervals

n	*z's*	*Co. eq.*	MUSCOD II		Algorithm 1	
			CPU-Time (ms)	*J*	CPU-Time (ms)	*J*
5	18	12	43	0.573117	188	0.568171
10	33	22	53	0.573080	290	0.572162
20	63	42	146	0.573527	350	0.573290
40	123	82	940	0.573544	480	0.573478
80	243	162	3620	0.573545	547	0.573528
160	483	322	21612	0.573545	735	0.573541

n: number of subintervals; *z's*: total number of variables; *Co. eq.*: total number of constraints; *J*: value of objective function.

Example 2: Optimal control of a continuous stirred tank reactor (CSTR): We consider a CSTR as shown in Fig. 5.

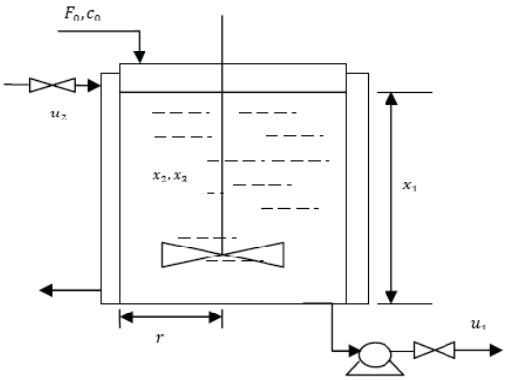


Fig. 5: CSTR example

An exothermic, irreversible, first order reaction A B occurs in the liquid phase and the temperature is regulated with external cooling. This highly nonlinear example is taken form Henson and Seborg (1997) or Pannocchia and Rawlings (2003) with the assumption that the level liquid is not constant. The constrained optimal control problem is formulated as follows:

$$\min_{x,u} \int_0^{t_f} [(x_1 - x_1^s)^2 + 100(x_2 - x_2^s)^2 + 0.1(u_1 - u_1^s)^2 + 0.1(u_2 - u_2^s)^2]dt \quad (21)$$

s.t.

$$\dot{x}_1 = \frac{F_0 - u_1}{\pi r^2}$$

$$\dot{x}_2 = \frac{F_0(c_0 - x_2)}{\pi r^2 x_1} - k_0 x_2 e^{-E/RT}$$

$$\dot{x}_3 = \frac{F_0(T_0 - x_3)}{\pi r^2 x_1} + \frac{-\Delta H}{\rho C_p} k_0 x_2 e^{-E/RT} + \frac{2U}{r\rho C_p}(u_2 - x_3)$$

$$x_1(0) = 0.659, \quad x_2(0) = 0.877 \text{ and } x_1(0) = 324.5$$

$$0.5 \le x_1 \le 2.5, \quad 0.8 \le x_2 \le 1.0$$

$$85 \le u_1 \le 115, \quad 299 \le u_2 \le 301$$

where x_1 is the level of the tank in meter, x_2 the product concentration in mol and x_3 the reaction temperature in (K), and the controls are u_1 and u_2 the outlet flow rate in (L/min) and coolant liquid temperature, respectively. In addition the inlet flow rate F_0 or the inlet concentration c_0 is acting as a disturbance to CSTR. The desired steady-state operating points: x_1^s, x_2^s, u_1^s and u_2^s are 0.659 meter, 0.877 mol/L, 100L/min and 300K, respectively. The model parameters in nominal conditions are shown in Table 2. We consider the operation case that at the tenth minute a disturbance enters the plant at a level of 0.05 mol/L on the inlet molar concentration c_0. A time horizon of $t_f = 50$ min is considered.

Table 2: Parameters of the CSTR

F_0	100 L/min	E/R	8750 K
T_0	305 K	U	915.6 $Wm^{-2}K^{-1}$
c_0	1.0 mol/L	ρ	1 kg/L
R	0.219	C_p	0.239 $Jg^{-1}K^{-1}$
k_0	0.219	ΔH	-5×10^4 J/mol

To solve problem (21) using the proposed algorithm we divide the time horizon into 50 subintervals and so that the number of resulted NLP will be 306 variables with 204 constraints, and the same PC is used to make the optimization. We used the IPOPT 3.4.0 to solve the NLP and NAG mark 8 to solve the Newton-Raphson equations and linear equation systems.

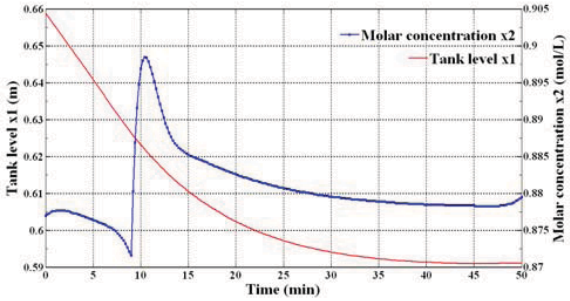


Fig. 6: The optimal output flow $x_1(t)$ and coolant temperature $x_2(t)$.

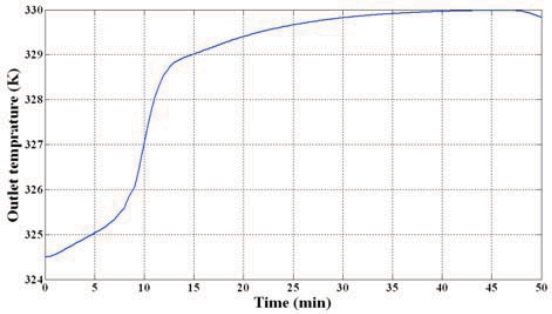


Fig. 7: The optimal outlet temperature $x_3(t)$.

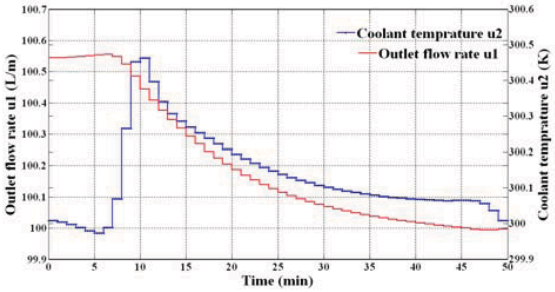


Fig. 8: The optimal control profiles $u_1(t)$ and $u_2(t)$.

Figures 6 and 7 show the optimal control profiles of the states $x_1(t)$, $x_1(t)$ and $x_3(t)$, respectively and Fig. 8 shows the optimal control profiles $u_1(t)$ and $u_2(t)$. The objective function value at the optimum is 0.9015886. Moreover, the algorithm was converged in 35 iterations and with the CPU-time in 0.954s. In comparison, this problem was also solved by Hong *et al.* (2006) using a quasi-sequential approach and it was converged in 16 iterations and 5.56 s of CPU time of SUN Ultra 10 Station with identical solutions.

6. CONCLUSIONS

In this paper we proposed a novel algorithm for NMPC. It is a combination of the multiple shooting, where the NLP problem will be handled, with the collocation method, where function values and gradients required in the NLP will be computed. We use piecewise constant for controls and the three-point collocation for states to parameterize the vector of optimization variables. The proposed algorithm has been realized in the framework of the numerical algorithm group (NAG) and IPOPT in the C/C++ environment. In addition, two demonstrative examples have been taken as case studies to show and compare the results from our algorithm and the well known MUSCOD II code. From these results it can be seen that the proposed algorithm is more efficient when a large-scale NMPC problem is to be solved. Stability and error control issues as well as practical applications of this algorithm will be considered in our future work.

7. ACKNOWLEDGEMENT

The financial support from the German Academic Exchange Service (DAAD) for this work is gratefully acknowledged.

8. REFERENCES

Biegler, L. T., Cervantes, A. M. and Wächter, A. (2002). Advances in simultaneous strategies for dynamic process optimization. *Chemical Engineering Science*. 4(57), 575–593.

Bock, H.G. and Plitt, K.J. (1984). A multiple shooting algorithm for direct solution of optimal control problems. *In: Proceedings 9th IFAC World Congress Budapest*. Pergamon Press, 243-247.

Bryson, A. E. and Ho, Y.C. (1975). *Applied Optimal control*, Hemisphere publication corporation, Levittown.

Diehl, M. (2001). Real-time optimization for large scale nonlinear processes. PhD Thesis, *University of Heidelberg*.

Diehl, M., Findeisen, R., Schwarzkopf, S., Uslu, I., Allgöwer, F., Bock, H.G., Gilles, E.D., and Schlöder, J.P. (2002a). An efficient algorithm for nonlinear model predictive control of large-scale systems. Part I: Description of the method. *At-Automatisierungstechnik.* 50(12), 557-567.

Diehl, M., Bock, H.G. Schlöder, J.P., Findeisen, R., Nagy, Z., and Allgöwer, F. (2002b) Real-time optimization and nonlinear model predictive control of processes governed by differential-algebraic equations. *Journal of Process Control.* 12(4), 577-585.

Diehl, M., Schäfer, A. and Leineweber, D. B. (2001). *MOSCOD II user manual*, Interdisciplinary Center for Scientific Computing (IWR), University of Heidelberg.

Diehl, M., Bock, H.G., Diedam, H. and Wieber P.B. (2006). Fast direct multiple shooting algorithms for optimal robot control. LNCIS, *Fast Motions in Biomechanics and Robotics*, 340, 65-93.

Finlayson, B. A. (1980). *Nonlinear analysis in chemical engineering*, McGraw-Hill, New York.

Gill, P. E., Murray, W. and Saunders, M. A. (2002). SNOPT: an SQP algorithm for large-scale constrained optimization. *SIAM J. Optim*. 12(1). 979–1006.

Gill, P. E., Murray, W. and Saunders, M. A. (2008). *User's Guide for SNOPT version 7: Software for large-scale nonlinear programming.* Department of Mathematics, University of California, San Diego.

Gatlu, G. and Zafiriou, E. (1992). Nonlinear quadratic dynamic matrix control with state estimation. *Ind. Eng. Chem. Res*. 31(4), 1096-1104.

Golub, G. H. and van Loan, C. F. (1996). *Matrix computations.* (3rd ed.). Johns Hopkins University Press. Baltimore.

Henson, MA., Seborg, DE. (1997). *Nonlinear Process Control. Upper Saddle River.* NJ, Prentice Hall.

Hong, W. R., Wang, S. Q., Li, P., Wozny, G., Biegler, L.T. (2006), A quasi-sequential approach to large-scale dynamic optimization problems, *AIChE Journal*, 52(1), 255-268.

Kirk, D. E. (1970). *Optimal Control*, Prentice Hall Inc, Englewood Cliffs.

Kraft, D. (1985). On converting optimal control problems into nonlinear programming problems. *In K. Schittkowski. Computational Mathematical Programming,* vol. F15 of NATO ASI, 261–280. Springer.

Leineweber, D. B. (1995). Analyse und Restrukturierung eines Verfahrens zur direkten Lösung von Optimal-Steuerungsproblem, The Theory of MUSCOD II in a Nutshell. Master Thesis, University of Heidelberg.

Lewis, F. L., and Syrmos, V. L. (1995). *Optimal Control Theory*, Wiley Interscience.

Li, P. (2007). Prozessoptimierung: Methoden, Anwendungen und Herausforderungen, *Chemie Ingenieur Technik*. 79(10), 1567-1580.

Numerical Algorithms Group Ltd. (2005), *NAG C Library Manual, Mark 8.* Oxford, UK.

Pannocchia, G., Rawlings, J., Disturbance models for offset-free model predictive control. *AIChE J*. 49, 426-437

Qin, S.J., Badgwell, T.A. (2003). A survey of industrial model predictive control technology, *Control Engineering Practice.* 11(7), 1096-1104.

Sargent, R.W.H. and Sullivan, G.R. (1977). The development of an efficient optimal control package. In J. Stoer, *Proceedings of the 8th IFIP Conference on Optimization Techniques,* Part 2. **7**(1978), 158-168. Springer. Heidelberg.

Schäfer, A., Kühl, P. Diehl, M., Schlöder, J. and Bock, H.G. (2007). Fast reduced multiple shooting methods for nonlinear model predictive control. *Chemical Engineering & Processing*, 46(11), 1200–1214.

Wächter, A. (2008), Introduction to IPOP: A tutorial for downloading, installing and using IPOPT. Department of Mathematical Sciences, IBM T.J. Watson Research Center, Yorktown Heights, NY.

Schäfer, A., Kühl, P. Diehl, M., Schlöder, J. and Bock, H.G. (2007). Fast reduced multiple shooting methods for nonlinear model predictive control. *Chemical Engineering & Processing*, 46(11), 1200–1214.

Robust Control of Yeast Fed-Batch Cultures for Productivity Enhancement

D. Coutinho [*,**] **L. Dewasme** [*] **A. Vande Wouwer** [*]

[*] *Service d'Automatique, Faculté Polytechnique de Mons, Boulevard Dolez 31, B-7000 Mons, Belgium (e-mails: Daniel.Coutinho(Alain.VandeWouwer,Laurent.Dewasme)@fpms.ac.be)*
[**] *On leave from the Group of Automation and Control Systems, Faculty of Engineering, Pontifícia Universidade Católica do Rio Grande do Sul, Av. Ipiranga 6681, Porto Alegre-RS, 90619-900 Brazil*

Abstract: This work proposes a robust control strategy for the optimizing control of fed-batch cultures of S. cerevisae. The process dynamics is characterized by a nonlinear kinetic model based on the bottleneck assumption and ethanol inhibition for a possible excess of substrate feeding. The control strategy is based on the feedback linearization technique, where the resulting free linear dynamics is designed so as to ensure a certain robustness to plant parameter variations. A feedforward loop achieves the correct critical substrate value, which is a function of the ethanol and oxygen in the culture medium. In addition, a robust Luenberger-like observer is designed taking plant parameter variations into account. Numerical experiments demonstrate the potential of the proposed approach as a tool for control design of fed-batch cultures.

Keywords: robust control, feedback linearisation, Luenberger observer, fermentation process.

1. INTRODUCTION

The culture of host recombinant micro-organisms is probably the only economical way of producing pharmaceutical biochemicals. The cell cultures or the culture of micro-organisms are basically operated in three different modes – batch, fed-batch and continuous. The fed-batch operation is popular in industrial practice, because it is advantageous from an operational and control point of view (Roeva and Tzonkov, 2005). In this mode of operation, the bioreactor is manipulated by controlling its feeding rate. The off line design of the optimal feeding profile in general does not give high productivity, since in open-loop an excess of substrate leads to the accumulation of by-products (ethanol for yeast and acetate for bacteria), which in turn yields an inhibition of the cell respiratory capacity.

To avoid high concentrations of inhibitory by-product, a closed-loop solution is in general applied leading to a wide diversity of approaches (Chen et al., 1995; Boskovic and Narendra, 1995; Hisbullah et al., 2002; Rocha et al., 2004; Renard and Vande Wouwer, 2008; Ignatova et al., 2008). Nevertheless, the closed-loop control optimization of yeast fed-batch process is still a challenging task for two main reasons. Firstly, the process kinetics is governed by highly nonlinear functions with uncertain model parameters. Secondly, there is a lack of reliable and low cost online sensors for the measurement of key state variables.

In the control context, many researchers are applying online algorithms to cope with time-varying model uncertainties by either adaptive control (Renard and Vande Wouwer, 2008; Dewasme and Vande Wouwer, 2008; Ignatova et al., 2008) or computational intelligence based algorithms (Rocha et al., 2004; Karakuzu et al., 2006). However, the use of online adaption schemes may lead to closed-loop instability in the presence of unmodeled dynamics. In this paper, we follow a different direction by applying the robust control theory to design a nonlinear controller (with a fixed parametrization) taking model uncertainties into account. The control strategy is based on the classical feedback linearizing technique which is widely applied in fermentation process (Bastin and Dochain, 1990). However, feedback linearizing control schemes are very sensitive to model uncertainties. To handle the lack of robustness, the resulting linear dynamics is designed in order not only to improve the overall performance but also to achieve robustness against model uncertainties.

On the other hand, complex control methods need in general full state information which in most of the situations is not practical. In this case, many approaches have been proposed in the process control literature to estimate some unavailable key states based on Luenberger observer (LO) and Kalman filter (KF) (Bastin and Dochain, 1990; Klockow et al., 2008). However, these state estimators are implemented iteratively (e.g., extended LO and KF) to deal with the nonlinearities exhibited in the fermentation dynamics making difficult the task of tuning the observer gain in order to achieve a nice convergence behaviour. In this paper, we propose a robust nonlinear observer for which a nonlinear static gain is designed to improve the estimation convergence as well as to cope with model uncertainties. The rest of this paper is as follows. Section 2 introduces the problem to be addressed in this paper. The control strategy is proposed in Section 3 and the robust observer design is derived in Section 4. Numerical experiments are carried out in Section 5 to validate the approach and Section 6 ends the paper.

2. PRELIMINARIES

The yeast strain *S. cerevisiae* presents a metabolism that is macroscopically described as follows (Bastin and Dochain, 1990):

$$\text{Substrate oxidation:} \quad S + k_5 O \xrightarrow{r_1} k_1 X + k_7 P \tag{1}$$

$$\text{Substrate fermentation:} \quad S \xrightarrow{r_2} k_2 X + k_4 E + k_8 P \tag{2}$$

$$\text{Ethanol oxidation:} \quad E + k_6 O \xrightarrow{r_3} k_3 X + k_9 P \tag{3}$$

where X, S, E, O and P are, respectively, the concentration in the culture medium of biomass, substrate (typically glucose), ethanol, dissolved oxygen and carbon dioxide. The k_i, $i = 1, \ldots, 9$, are the constant yield coefficients and the r_i, $i = 1, 2, 3$, are the specific growth rates. We model these rates by the following discontinuous functions:

$$r_1 = \min\{r_S, k_5^{-1} r_O\} \tag{4}$$

$$r_2 = \max\{0, r_S - k_5^{-1} r_O\} \tag{5}$$

$$r_3 = \max\left\{0, \frac{r_O - k_5 r_S}{k_6} \cdot \frac{E}{E + K_E}\right\} \tag{6}$$

where the kinetic terms related to the substrate consumption r_S, the oxidative or respiratory capacity r_O and the ethanol oxidative rate r_E are represented as follows

$$r_S = \mu_S \, \frac{S}{S + K_S} \tag{7}$$

$$r_O = \mu_O \, \frac{O}{O + K_O} \cdot \frac{K_{i_E}}{K_{i_E} + E} \tag{8}$$

$$r_E = \mu_E \, \frac{E}{E + K_E} \tag{9}$$

with the constants μ_S, μ_O and μ_E being the maximal values of the specific growth rates and K_S, K_O and K_E expressing the saturation of the respective elements. Note that we are taking the effect of ethanol on the cells growth into account by considering the inhibition ethanol constant K_{i_E} in (8).

The component-wise mass balances of the above reaction scheme lead to the following state-space representation (Dewasme and Vande Wouwer, 2008)

$$\dot{x} = K r(x) x_1 + A x - u x + B(u) \tag{10}$$

where $x = [\, x_1 \; x_2 \; x_3 \; x_4 \; x_5 \; x_6 \,]' = [\, X \; S \; E \; O \; P \; V \,]'$ is the state vector with $x_6 = V$ being the culture medium volume, $r(x) = [\, r_1 \; r_2 \; r_3 \,]'$ is the vector of reaction rates, and $u = F_{in}/x_6$ is the control input (the dilution rate) with F_{in} denoting the inlet feed rate. The matrices K and A, and the vector function $B(\cdot)$ are given by:

$$K = \begin{bmatrix} k_1 & k_2 & k_3 \\ -1 & -1 & 0 \\ 0 & k_4 & -1 \\ -k_5 & 0 & -k_6 \\ k_7 & k_8 & k_9 \\ 0 & 0 & 0 \end{bmatrix}, \; B(u) = \begin{bmatrix} 0 \\ S_{in} \; u \\ 0 \\ k_L a \; O_{sat} \\ k_L a \; P_{sat} \\ 0 \end{bmatrix}, \tag{11}$$

$$A = \begin{bmatrix} 0_3 & 0_{3\times 2} & 0_{3\times 1} \\ 0 & -k_L a \; I_2 & 0 \\ 0_{1\times 3} & 0_{1\times 2} & 0 \end{bmatrix},$$

where $k_L a$ is the volumetric transfer coefficient, S_{in} is the feeding substrate concentration, and O_{sat} and P_{sat} are respectively the saturations of dissolved oxygen and carbon dioxide concentrations.

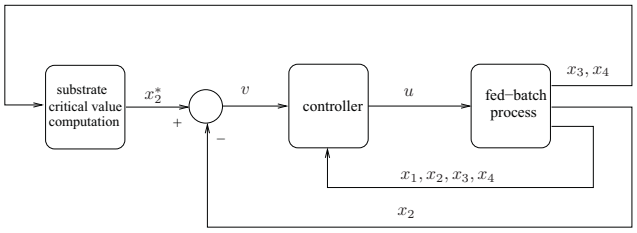


Fig. 1. Control Scheme for Optimal Operation Regime.

To analyze the biomass productivity, we recall the Sonnleitner's bottleneck assumption (Sonnleitner and Käppeli, 1986) which states that during a culture the yeast cells are likely to change their metabolism because of limited respiratory capacity. When the substrate concentration is large, the yeast cells produce ethanol (respiro-fermentative regime). If the substrate concentration becomes small, the available substrate (and possibly the ethanol) are oxidized (respirative regime). Thus, the optimal operating point to maximize the biomass productivity is at the boundary of the two regimes (Valentinotti et al., 2004), i.e., when the fermentation and oxidation reaction rates are equal to zero. Hence, the optimal operating point can be easily computed through the equality $r_O = k_5 r_S$ leading to the following equation

$$x_2^* = \frac{K_S \; r_O}{k_5 \mu_S - r_O} \tag{12}$$

where x_2^* refers to the substrate critical value.

In view of (8), we note that the operating point x_2^* is in fact a nonlinear function of x_3 and x_4. To simplify the control problem, many references either consider a constant set-point (Klockow et al., 2008) or alternatively choose a sub-optimal solution by imposing a low-level of ethanol concentration (Renard and Vande Wouwer, 2008).

3. CONTROL STRATEGY

In this paper, we aim at maintaining the system as close as possible to its optimal operating condition. To this end, we have to determine on-line the value of x_2^* and design a controller such that x_2 tracks approximately x_2^*. In addition, to simplify the analysis, we suppose in this section that all states are available on-line for feedback. For practical purposes, a nonlinear observer is proposed in the next section to estimate some state variables which are difficult to measure.

The proposed control scheme is illustrated in Figure 1. The internal feedback loops correspond to a standard feedback linearizing controller, where the free linear dynamics is designed to give a good tracking response as well as to assure a certain level of robustness against plant parameter variations. The external feedforward loop is to compute on-line the substrate critical level. We stress that instead of computing an adaptive controller to handle plant parameter variations (as, e.g., Dewasme and Vande Wouwer (2008)), we design a fixed controller that will have a guaranteed performance in the admissible parameter space.

To control the substrate level, consider the following dynamics for x_2 taken from (10)

$$\dot{x}_2 = -(r_1 + r_2)x_1 + (S_{in} - x_2)u \tag{13}$$

where r_1 and r_2 are nonlinear functions of x_2, x_3 and x_4 as given by (4) and (5). With respect to the above system dynamics, we assume that the values of $x_1, \ldots, x_6$ are bounded to a given polytopic region $\mathcal{X}$ with know vertices, that is, $x \in \mathcal{X} \subset \mathbb{R}^6$.

A feedback linearizing control law can be easily derived:

$$u = \frac{F_{in}}{x_6} = \frac{1}{S_{in} - x_2}((\tilde{r}_1 + \tilde{r}_2)x_1 + v) \tag{14}$$

where $\tilde{r}_1$ and $\tilde{r}_2$ are respectively the nominal values of r_1 and r_2, which may vary due to parameter variations, and v is the new input of the resulting linearized system.

In view of (13) and (14), we obtain the following dynamics for x_2

$$\dot{x}_2 = v - (e_{r_1} + e_{r_2})x_1 \tag{15}$$

where $e_{r_1} := r_1 - \tilde{r}_1$ and $e_{r_2} := r_2 - \tilde{r}_2$ are nonlinear functions of (x_2, x_3, x_4) representing possible inexact cancelations of nonlinear terms due to uncertain model parameters.

Borrowing the ideas of the *Quasi-LPV* approach (Leith and Leithead, 2000), we bound the term $e_{r_1} + e_{r_2}$ by a time-varying parameter $\delta = \delta(t)$ which is supposed to belong to a known set $\Delta := \{\delta : \underline{\delta} \leq \delta \leq \overline{\delta}\}$ with $\underline{\delta}$ and $\overline{\delta}$ respectively representing the minimal and maximal admissible uncertainty.

To approximately track the time-varying reference signal x_2^*, we consider the following additional control loop

$$v = \lambda(x_2^* - x_2) \tag{16}$$

where $\lambda \in \mathbb{R}$ is a free parameter to be designed.

In this paper, we design the parameter λ to ensure some robustness and a certain tracking performance to the overall closed loop system. To this end, we model the closed loop system as follows

$$\mathcal{M} : \left\{ \begin{array}{rcl} \dot{x}_2 &=& -\lambda x_2 + a(\lambda, \delta)w \\ z &=& -\ x_2\ +\ c\ w\ ,\ \delta \in \Delta \end{array} \right. \tag{17}$$

where $w = [\, x_2^* \quad x_1 \,]' \subset \mathcal{L}_{2,[0,T]}$ is a disturbance input to the system $\mathcal{M}$, $z = x_2^* - x_2$ the performance output and

$$a(\lambda, \delta) = [\, \lambda \quad -\delta \,]\ ,\ c = [\, 1 \quad 0 \,]\ .$$

Now, consider the following definition for the finite horizon $\mathcal{L}_2$-gain of system $\mathcal{M}$:

$$\|\mathcal{M}_{wz}\|_{\infty,[0,T]} = \sup_{\delta \in \Delta, 0 \neq w \subset \mathcal{L}_{2,[0,T]}} \frac{\|z\|_{2,[0,T]}}{\|w\|_{2,[0,T]}} \tag{18}$$

Thus, we design the parameter λ based on the $\mathcal{H}_\infty$ control theory (Skogestad and Postlethwaite, 2001). In other words, we solve the following optimization problem

$$\min_{\lambda, \delta \in \Delta}\ \gamma : \|\mathcal{M}_{wz}\|_{\infty,[0,T]} \leq \gamma \tag{19}$$

while ensuring the robust stability of system (17).

Note 1. The parameter λ can be easily obtained through the LMI framework either via a quadratic Lyapunov function (Boyd et al., 1994) or a parameter dependent one (de Souza et al., 2000) if we assume $\dot{\delta}$ is also bounded, since we can easily perform a line search on λ. □

4. ROBUST OBSERVER

To implement the control law proposed in the latter section, we have to measure several state variables such as X, S, E and O. In spite of existing specific probes to measure all these signals on-line, some sensors can be quite expensive and are not always available in a practical set-up. Particularly, in the proposed control strategy, we are dealing with very low levels of substrate (glucose) and ethanol concentrations making their measurements expensive and inaccurate.

Alternatively, we propose a robust Luenberger-like nonlinear observer to estimate the substrate and ethanol concentration levels from the measurement of $x_1 = X$, $x_4 = O$, $x_5 = P$ and the dilution rate $u = F_{in}/x_6$. As we are dealing with a nonlinear system, the exponential observability property of the system is state dependent (Bastin and Dochain, 1990). In other words, for large estimation errors, the observer may diverge from the system operating point since the exponential observability is lost. To overcome this problem, we assume the initial conditions $x_2(0)$ and $x_3(0)$, which are respectively the initial substrate and ethanol concentration levels, are partially known (likely through inaccurate off-line measurements).

Firstly, we model the reaction rates by the following uncertain functions:

$$r_i(x) \cong r_i(\theta_i) = \alpha_i(1 + \beta_i\theta_i)\ ,\ \theta_i \in [-1, 1] \tag{20}$$

where, for $i = 1, 2, 3$, α_i is the steady-state value of r_i, θ_i is an uncertain time-varying parameter which models the displacement of r_i from its steady-state regime and also a possible inaccuracy on the system parameters, and β_i is a given constant added in light of the unitary normalization of the uncertain parameter space. Then, we propose the following state space representation for the observer

$$\left\{ \begin{array}{l} \dot{\hat{x}} = K\hat{r}\hat{x}_1 + \hat{A}(u)\hat{x} + \hat{B}(u, y) + L(y, u)(y - \hat{y}) \\ \hat{y} = C_y\hat{x} \\ \hat{z} = C_z\hat{x} \end{array} \right. \tag{21}$$

where $\hat{x} \in \mathbb{R}^6$ is the state estimation, $y = C_y x$ is the on-line measurement, $\hat{y}$ is the measurement estimation, $\hat{z}$ is the signal to be estimate, K is as in (11), $L(y, u) \in \mathbb{R}^{6\times 4}$ is a nonlinear matrix function of y and u to be determined, $\hat{r}$ is as defined in (24), and

$$\begin{array}{c} \hat{A}(u) = -\mathrm{diag}\{0, u, u, k_L a, k_L a, 0\} \\ \hat{B}(u, y) = [\, -x_1 u \quad S_{in}u \quad 0 \quad (k_L a O_{sat} - x_4 u) \\ \qquad\qquad (k_L a P_{sat} - x_5 u) \quad -x_6 u \,]' \end{array} \tag{22}$$

$$C_y = \begin{bmatrix} 1&0&0&0&0&0 \\ 0&0&0&1&0&0 \\ 0&0&0&0&1&0 \\ 0&0&0&0&0&1 \end{bmatrix}\ ,\ C_z = \begin{bmatrix} 0&1&0&0&0&0 \\ 0&0&1&0&0&0 \end{bmatrix}\ .$$

Accordingly to (20), we define the estimates of $r_i(\theta_i)$ as follows

$$\hat{r}_i := \alpha_i \tag{23}$$

where, for $i = 1, 2, 3$, $\hat{r}_i$ is the estimate of the approximate reaction rates.

Now, considering the following notation

$$r(\theta) = \begin{bmatrix} \alpha_1(1+\beta_1\theta_1) \\ \alpha_2(1+\beta_2\theta_2) \\ \alpha_3(1+\beta_3\theta_3) \end{bmatrix}\ ,\ \theta = \begin{bmatrix} \theta_1 \\ \theta_2 \\ \theta_3 \end{bmatrix}\ ,\ \hat{r} = \begin{bmatrix} \alpha_1 \\ \alpha_2 \\ \alpha_3 \end{bmatrix}\ , \tag{24}$$

we can approximate the error dynamics as follows

$$\begin{aligned}\dot{e} &\cong (K\hat{r}N_r + \hat{A}(u) - L(y,u)C_y)e + K(r(\theta) - \hat{r})x_1 \\ &\cong (K\hat{r}N_r + \hat{A}(u) - L(y,u)C_y)e + K\Omega(x_1)\theta\end{aligned}$$

where $N_r = [\,1\ 0\ \cdots\ 0\,]$, $\theta \in \Theta := \{\theta \in \mathbb{R}^3 : |\theta_i| \leq 1\ ,\ i = 1,2,3\}$ and $\Omega(x_1) = x_1 \cdot \mathrm{diag}\{\alpha_1\beta_1, \alpha_2\beta_2, \alpha_3\beta_3\}$.

In light of the above developments, we can pose the problem of determining $L(y,u)$ in an ℓ_1 optimal control setting (Dahleh and Diaz-Bobillo, 1995). To this end, consider the following error dynamics representation:

$$\mathcal{E} : \begin{cases} \dot{e} &= A_e e + B_e \theta \\ z_e &= C_z e\ ,\ \|\theta\|_\infty \leq 1 \end{cases} \tag{25}$$

where θ is an energy-peak bounded disturbance signal, z_e the estimation error to be minimized and

$$A_e = K\hat{r}N_r + \hat{A}(u) - L(y,u)C_y\ ,\ B_e = K\Omega(x_1)\ .$$

In this paper, we consider the following definition for the ℓ_1-norm of system (25):

$$\|\mathcal{E}_{\theta z_e}\|_1 = \sup_{\substack{e \in \mathbb{E},\, e(0)=0 \\ \|\theta\|_\infty \leq 1}} \|z_e\|_\infty \tag{26}$$

where $\mathbb{E} := \{e : V(e) \leq 1\}$ is an estimate of the reachable set and $V(e)$ is a Lyapunov function for system $\mathcal{E}$, which guarantees the system internal stability.

An upper-bound σ on $\|\mathcal{E}_{\theta\ e_z}\|_1^2$ can be determined via the following optimization problem (Nagpal et al., 1994)

$$\min_{\substack{V(e),\eta \\ e \in \mathbb{E}}} \sigma : \begin{cases} V(e) > 0\ ,\ \eta > 0 \\ \dot{V}(e) + \eta(V(e) - \theta'\theta) < 0 \\ V(e) - \dfrac{z_e' z_e}{\sigma} \geq 0 \end{cases} \tag{27}$$

Notice the set invariance property of $\mathbb{E}$ is guaranteed for zero initial conditions and the constraints on (27) may not hold when $e(0) \neq 0$. As a result, the error state trajectory may leave $\mathbb{E}$ and do not return since the state observer is nonlinear and the stability properties are not necessarily global. In this paper, we assume the initial error is sufficiently close to zero such that $\mathbb{E}$ is attractive.

5. NUMERICAL EXPERIMENTS

In this section, we perform several numerical experiments considering small-scale culture conditions. In particular, we borrow the 20 $[l]$ bioreactor studied in (Dewasme and Vande Wouwer, 2008), where the initial and operating conditions are:

$$\begin{gathered} x_1(0) = 0.4\ [g/l]\ ,\ x_2(0) = 0.5\ [g/l]\ ,\ x_3(0) = 3\ [g/l]\ , \\ x_4(0) = O_{sat} = 0.035\ [g/l]\ ,\ x_5(0) = P_{sat} = 1.286\ [g/l]\ , \\ x_6(0) = 6.8\ [l] \quad \text{and} \quad S_{in} = 350\ [g/l]\ . \end{gathered}$$

We study two different scenarios. Firstly, supposing the state variables are available online for feedback, we design the robust linearizing feedback controller proposed in Section 3 aiming for tracking as close as possible the estimation of the substrate critical value. In this setup, we consider a noisy ethanol measurement, since the level of ethanol is likely to be very close to zero making difficult its measurement. Secondly, we design a robust observer to estimate the substrate and ethanol concentration levels, which in the proposed strategy are very low and difficult to measure in current practice, applying the result proposed in Section 4. In this case, we analyze the observer robustness and verify the set of initial conditions in which the convergence properties hold.

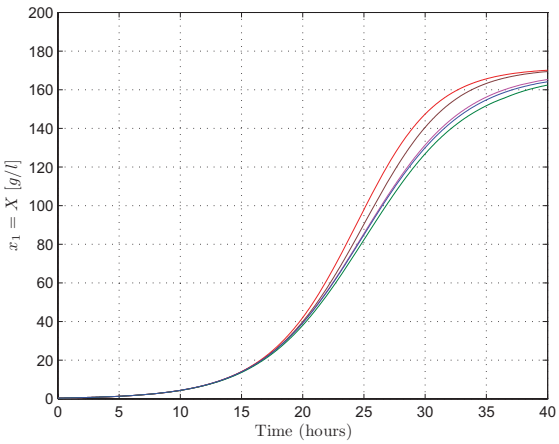


Fig. 2. Biomass concentration – state feedback case.

5.1 State Feedback

We refer to state feedback the control law proposed in (14) and (16), where x_1, x_2, x_3, x_4 and u are available online. To design the parameter λ in (16) via the optimization problem (19), we suppose the parameters K_S, K_E, K_O and K_{i_E} may vary $\pm 20\%$ from their nominal values. Simulating the operating conditions of the control strategy in (14), we may infer that $\overline{\delta} = -\underline{\delta} = 1.0$, which in light of (17) and (19) yields $\lambda = 44.8511$.

Figures 2 to 4 show the closed-loop response of biomass x_1, substrate x_2 and ethanol x_3 concentrations, for five different values of K_S, K_E, K_O and K_{i_E} (which were randomly chosen). In all simulations, we have added a white noise on the ethanol concentration measurement with a maximal amplitude of ± 0.25 $[g/l]$. Notice in all cases the biomass productivity does not significantly vary against parameter uncertainty and noise measurement.

5.2 Output Feedback

In order to design the state observer as proposed in Section 4, we have considered

$$\begin{gathered} \alpha_1 = 3.2 \times 10^{-5}\ ,\ \alpha_2 = 1.3 \times 10^{-6}\ ,\ \alpha_3 = 4 \times 10^{-7}\ , \\ \beta_1 = \beta_2 = \beta_3 = 1\ ,\ x_1 \in [0.4, 180]\ ,\ F_{in} \in [10^{-7}, 10^{-4}]\ , \end{gathered}$$

which are obtained from the noiseless simulations of the state-feedback case.

We can compute the observer gain through the LMI framework, see for instance (Coutinho et al., 2005). Assuming that u is available online, we have chosen an observer gain as follows:

$$L(y,u) = L(u) = L_0 + uL_1\ ,$$

where L_0 and L_1 are constant matrices to be determined. In addition, to simplify the computations, we constraint the Lyapunov function to be quadratic, i.e., $V(e) = e'Pe$ with $P = P' > 0$.

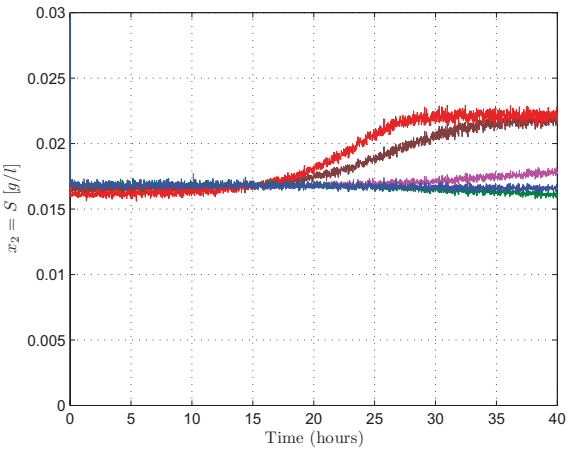


Fig. 3. Substrate concentration – state feedback case.

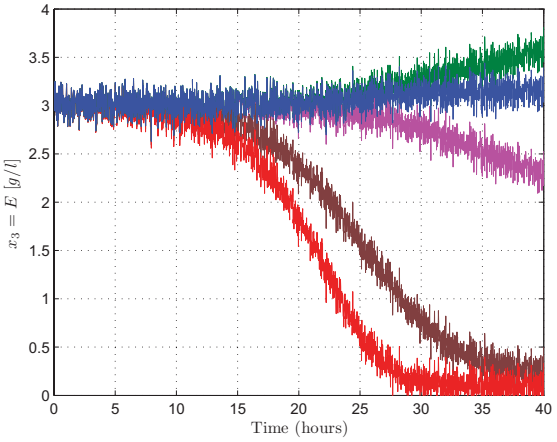


Fig. 4. Ethanol concentration – state feedback case.

Thus, solving (27) for all $(x_1, u) \in \mathcal{V}([0.4, 180] \times [10^{-7}, 10^{-4}])$ with the parametrization $Q(u) = PL(u)$ and a line search on η, we obtain the following matrices

$$L_0 = 10^6 \times \begin{bmatrix} 0.267 & -0.380 & 1.006 & 0.000 \\ -0.208 & 1.421 & -3.759 & 0.000 \\ 0.058 & -0.341 & 0.903 & 0.000 \\ -0.041 & 0.283 & -0.747 & 0.000 \\ 0.109 & -0.747 & 1.978 & 0.000 \\ 0.000 & 0.000 & 0.000 & -2\times 10^{-7} \end{bmatrix}$$

$$L_1 = 10^2 \times \begin{bmatrix} 0.143 & -0.256 & 0.679 & 0.000 \\ -0.172 & 1.217 & -3.220 & 0.000 \\ 0.045 & -0.290 & 0.768 & 0.000 \\ -0.034 & 0.241 & -0.639 & 0.000 \\ 0.090 & -0.639 & 1.690 & 0.000 \\ 0.000 & 0.000 & 0.000 & 0.000 \end{bmatrix}$$

where $\mathcal{V}(\cdot)$ stands for the set of vertices of $(\cdot)$.

From several simulations, the observer initial conditions that guarantee the stability of the error system are as follows

$$\hat{x}_2(0) = x_2(0) \pm 50\% \ , \ \hat{x}_3(0) = x_3(0) \pm 50\% \ . \qquad (28)$$

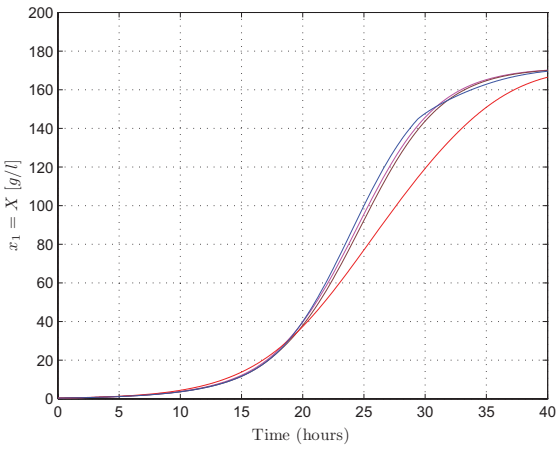


Fig. 5. Biomass concentration – output feedback case.

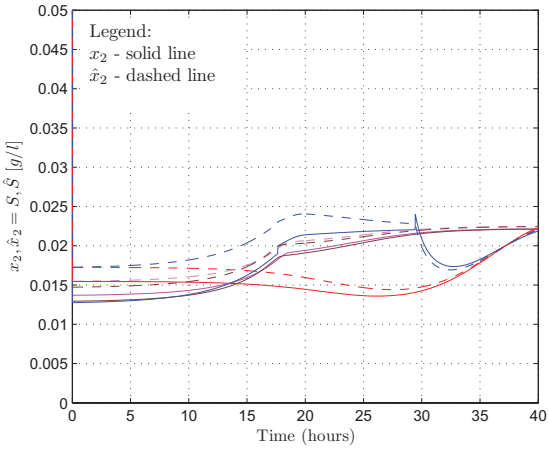


Fig. 6. Substrate concentration – output feedback case.

To test the output feedback closed-loop performance, we carried out several simulations for randomly chosen values of K_S, K_E, K_O, K_{iE} and $\hat{x}_2(0), \hat{x}_3(0)$ from the admissible parameter space leading to the results detailed in Figures 5, 6 and 7.

5.3 Remarks and Future Research

The simulations indicate that the overall performance of the biomass concentration productivity is robust against uncertainties on model parameters and some initial condition estimates. The biomass productivity is similar to the one obtained in (Dewasme and Vande Wouwer, 2008), where an adaptive control is applied for a similar setup, but the proposed approach achieved a better transient performance. However, the ethanol concentration level does not always converge to zero indicating an error on the estimation of x_2^*. Notice we determine x_2^* from (12) which is a function of some partially known parameters. Further developments are needed to improve the estimation of the substrate concentration critical level.

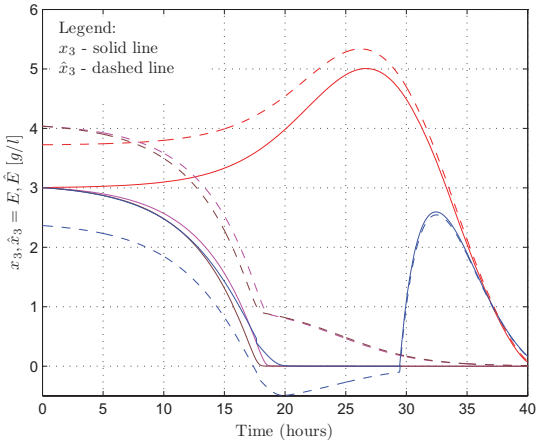


Fig. 7. Ethanol concentration – output feedback case.

6. CONCLUSION

This paper has proposed a robust control strategy to optimize the production of yeast cultures in fed-batch operation. Firstly, assuming full state information, a robust controller is designed for ensuring a guaranteed performance in spite of parameter uncertainty. Then, a nonlinear robust observer is derived in order to estimate the states that are not available online for feedback. Numerical examples have demonstrated the applicability of the proposed approach to control yeast fed-batch fermentation processes.

ACKNOWLEDGEMENTS

This paper presents research results of the Belgian Network DYSCO (Dynamical Systems, Control, and Optimization), funded by the Interuniversity Attraction Poles Programme, initiated by the Belgian Federal Science Policy Office (BELSPO). The scientific responsibility rests with its authors. D. Coutinho is beneficiary of a fellowship granted by BELSPO.

REFERENCES

Bastin, G. and Dochain, D. (1990). *On-line Estimation and Adaptive Control of Bioreactors*. Elsevier, Amsterdam, The Netherlands.

Boskovic, J. and Narendra, K. (1995). Comparion of linear, nonlinear and neural network-based adaptive controllers for a class of fed-batch fermentation processes. *Automatica*, 31, 817–840.

Boyd, S., El-Ghaoui, L., Feron, E., and Balakrishnan, V. (1994). *Linear Matrix Inequalities in System and Control Theory*. SIAM, Philadelphia, PA.

Chen, L., Bastin, G., and van Breusegem, V. (1995). A case study of adaptive nonlinear regulation of fed-batch biological reactors. *Automatica*, 31, 55–65.

Coutinho, D., Curcio, M., Mladic, J., and Bazanella, A. (2005). Robust Observer Design for a Class of Nonlinear Systems. In *Proc. 44th IEEE Conf. Decision Contr. and European Contr. Conf.*, 2640–2645. Seville, Spain.

Dahleh, M. and Diaz-Bobillo, I. (1995). *Control of Uncertain Systems: A Linear Programming Approach*. Prentice-Hall, Englewwod-Cliffs, NJ.

de Souza, C., Trofino, A., and de Oliveira, J. (2000). Robust H_∞ Control of Uncertain Linear Systems via Parameter-Dependent Lyapunov Functions. In *Proc. 39th IEEE Conf. Decision and Contr.*, 3194–3199. Sydney, Australia.

Dewasme, L. and Vande Wouwer, A. (2008). Adaptive extremum-seeking control applied to productivity optimization in yeast fed-batch cultures. In *Proc. 17th IFAC World Congress*, 9713–9718. Seoul, Korea.

Hisbullah, M., Hussain, K., and Ramachandran, K. (2002). Comparative evaluation of various control schemes for fed-batch fermentation. *Bioprocess and Biosystems Engineering*, 24, 309–318.

Ignatova, M., Lyubenova, V., Garcia, M., Vilas, C., and Alonso, A. (2008). Indirect adpative linearizing control of a class of bioprocess – Estimator tuning procedure. *Journal of Process Control*, 18, 27–35.

Karakuzu, C., Türker, M., and Öztürk, S. (2006). Modelling, on-line state estimation and fuzzy control of production scale of fed-batch baker's yeast fermentation. *Control Engineering Practice*, 14, 959–974.

Klockow, C., Hüll, D., and Hitzmann, B. (2008). Model based substrate set point control of yeast cultivation processes based on FIA measurements. *Analytica Chimica Acta*, 623, 30–37.

Leith, D. and Leithead, W. (2000). Survey of gain-scheduling analysis and design. *International Journal of Control*, 73, 1001–1025.

Nagpal, K., Abedor, J., and Poola, K. (1994). An LMI Approach to Peak-to-Peak Gain Minimization: Filtering and Control. In *Proc. Am. Contr. Conf.*, 742–746. Baltimore, MA.

Renard, F. and Vande Wouwer, A. (2008). Robust adaptive control of yeast fed-batch cultures. *Computers and Chemical Engineering*, 32, 1238–1248.

Rocha, M., Neves, J., Rocha, I., and Ferreira, E. (2004). Evolutionary algorithms for optimal control in fed-batch fermentation processes. In G.R. et al (ed.), *Applications of Evolutionary Computing - LNCS*, volume 3005, 84–93. Springer-Verlag, Berlin, Germany.

Roeva, O. and Tzonkov, S. (2005). Optimal Feed Rate Control of Escherichia coli Fed-batch Fermentation. *Bioautomation*, 2, 30–36.

Skogestad, S. and Postlethwaite, I. (2001). *Multivariable Feedback Control – Analysis and Design*. John Willey & Sons, New York, NJ.

Sonnleitner, B. and Käppeli, O. (1986). Growth of Saccaharomyces cerevisiae is controlled by its limited respiratory capacity: Formulation and verification of a hypothesis. *Biotechnology and Bioengineering*, 28(6), 927–937.

Valentinotti, S., Srinivasan, B., Holmberg, U., and Bonvin, D. (2004). An Optimal Operating Strategy for Fed-Batch Fermentations by Feeding the Overflow Metabolite. In *Proc. IFAC Int'l Symp. Advanced Contr. Chemical Processes – ADCHEM'03*. Kowloon, Hong Kong.

Human Operator Based Fuzzy Intuitive Controllers Tuned with Genetic Algorithms

Filipe Leandro de F. Barbosa*, Ming Tham, Jie Zhang**, André Domingues Quelhas***

**Petrobras SA – Petróleo Brasileiro SA – Rio de Janeiro – RJ, Brazil*
Rua Visconde de Duprat s/n – 8o andar, CEP: 20211-230 – Rio de Janeiro – RJ, Brazil
(Tel: +55 21 3487-3474; e-mail: filipeleandro@petrobras.com.br, quelhas@petrobras.com.br)
*** School of Chemical Engineering and Advanced Materials, Newcastle University,*
Newcastle upon Tyne NE1 7RU, UK (e-mail: ming.tham@ncl.ac.uk, jie.zhang@newcastle.ac.uk)

Abstract: A recent study (Desborough and Miller, 2001) revealed that a great majority of the control loops that operate in industry use the PID (Proportional-Integral-Derivative) controllers. Furthermore, the study has shown that more than one third of these loops were switched to manual for a considerable period of time, indicating poor behaviour of the controllers' performance. As was also reported, the gap between the industrial practice and the process control theory remains unchanged over the years, indicating that industry is looking for simple and easy to use technologies. The present research offers an alternative control scheme that intends to be a step towards introducing a new technology for practical implementation in industry. The controller is developed aiming to emulate human operators' actions when manually controlling SISO systems, subject to disturbances. The developed control scheme is based on an intuitive hypothetical model that describes the way human operators (HO) act in a manual control loop, generating the Human Operator Based Intuitive Controller (HOBIC). Since human operators typically use vague terms when describing control actions, it is natural to use fuzzy logic to express manual control actions. The HOBIC is then extended using the Fuzzy Logic theory. Membership functions within Fuzzy-HOBIC are tuned using a genetic algorithm (GA). The tuning does not require a process model. It is based on historical process operation data containing manual operation actions from experienced operators. The traditional GA is modified to cope with real valued optimisation variables and their constraints. Results show that the hypothetic model created for the HO's actions is appropriate, since the generated control actions by the HOBIC and Fuzzy-HOBIC can approximate those of human operators. The control signal generated has the same discontinuous nature of the HO's one.

Keywords: advanced control strategy, human operator model, auto-tuning, fuzzy logic, genetic algorithm

1. INTRODUCTION

Process Control theory has been well explored over decades to, at first, automate the manual control loops and thereafter, to maintain and improve them. However, it is observed that there is a gap between the Industry and the state of the art theory (Desborough and Miller, 2001). Industry presents, very often, a high resistance to put into practice what was recently developed. As a result, the vast majority of the control loops observed operates using traditional PID (Proportional-Integral-Derivative) controllers. Such kinds of controllers are very simple to apply and understand, when compared to other more advanced approaches.

However, recent studies have revealed that a great part of those PID control loops are having a poor behaviour or are operating in manual. As an example, Desborough and Miller (2001) published a survey showing that 97% of the regulatory controllers from over 11,000 control loops of refining, chemicals, pulp and paper industries use the PID strategy. Furthermore, this work mentions that only one third of these controllers provide an acceptable performance. It is also commented that this fact is in accordance with the work of Bialkoski (1993).

This leads to the challenge of investigating those control loops to verify if they can be improved by simply reviewing the applied PID strategies or, in some cases, by studying the applicability of new techniques. Nevertheless, being aware of Industry's inertia to new developments, the challenge is even greater: to develop a control strategy with a great practical appeal, so it can be easy to understand and apply.

In the authors' experience, very often, newly developed control strategies are difficult to understand by the operators, the end users of the controllers. Hence, at the first indication of bad behaviour of the loop, the operators simply switch the controller to manual and if the process engineer does not verify what caused the problem, the operator normally will not turn it into automatic again. This will reinforce the statistics of loops in manual in Industry and quickly put the new technology into disrepute.

The review by Desborough and Miller (2001) also explicitly shows that 36% of the analysed PID loops are operating in manual for at least 30% of the observed data (5,000 samples

at the dominant system time constant). Hence, almost 4,000 controllers have been switched to manual for a considerable amount of time.

This work has the objective of developing a technique which is easily understandable by the operators. By doing so, the end users of the control loop will tend to support the idea and maintain the controller in automatic.

One way of getting support of the operators is to make the control loop behave as if the loop is in manual. In other words, the controller has to give actions to the final control element in the same way as the operator would do, if he was controlling the loop. If the new controller manages to mimic this behaviour, it is less likely that the operator will switch the controller to manual.

Nevertheless, trying to emulate the operator's actions in a control scheme is not something new. Until 1966, over 200 works related to this subject were published, according to Costello and Higgins (1966).

The great majority of the research on Human Operator (HO) modelling in the past was for application in mechanical systems, such as aircrafts and vehicles dynamics (Kleinman, Baron and Levison, 1970). Investigating more recent papers in this field, it can be noticed that this area of modelling the human behaviour for applications in control systems still attracts the researchers' attention as can be seen by the works of Enab (1995); Zapata, Galvão and Yoneyama (1999); Ertugrul and Hizal (2005). However, there is still a lack of real applications of such control technique in the Process Control field.

The present research aims to construct a control system with direct application in the continuous Process Control Industry, also by modelling Human Operator actions. However, it is different from the vast majority of the previous works in this modelling field. Due to the fact that the system dominant time-constants in the Process Control area are, in general, greater than in the mechanical systems, the concern about the HO's reaction time becomes negligible. Hence, the HO modelling techniques applied by Kleinman, Baron and Levison (1970a, b) are not suitable for sluggish Process Control applications. In the same manner, Zapata, Galvão and Yoneyama (1999) presented a mechanical application where the system time-constant had the same order of magnitude of the HO responses. As a result, an ARMA (autoregressive with moving average) model for the HO had to be identified to smooth the operator actions, considered to be noisy and less consistent than the ARMA model ones.

Another important issue to be discussed is the implementation strategy that the recent works used to model the HO control actions. They applied a model-free type technique. In other words, the model was extracted based upon input-output data, either by using Neural Networks approach (Enab, 1995), Neuro-Fuzzy techniques (Ertugrul and Hizal, 2005) or simply by extracting Fuzzy rules directly (Zapata, Galvão and T. Yoneyama, 1999).

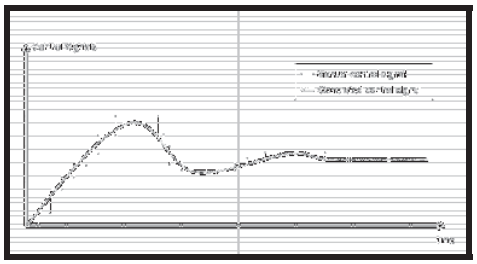


Fig. 1: Fictitious representation of a Manual control signal and a generated control signal using a model-free approach.

Although the model-free approach is able to approximate the HO behaviour, as the results of these works show, it fails to present a clear and easily understandable description of how the HO behaves and which rule system it uses to generate the actions. Even when applying Fuzzy Logic (FL) technique directly, as done by the work of Zapata, Galvão and T. Yoneyama (1999), the model-free approach generated a set of 15625 rules, which is quite difficult to understand and maintain in a practical application.

In the present work, a model-based approach is applied using the FL theory. Hence, the number of generated Fuzzy rules is expected to be much less than when using the model-free approach, and therefore easier to understand and apply in the Process Industry.

The work developed by Enab (1995) is of particular interest because it was related specifically to Process Control. The application presented was the control of the level in a tank, which has a nonlinear behaviour. This paper shows that the manual operation can be approximated using a Neural Networks approach. However, the generated control signal is continuous, compared with a "stair-like" manual signal, as can be seen by Fig.1. The difference in the signal's nature is clear. On the other hand, a FL model-based approach would be able to produce a "stair-like" signal, if the proposed rules that comprise the Human Operator model are designed to perform this task. Nevertheless, one disadvantage of the FL model-based system is that the resultant Fuzzy Logic Controller (FLC) will need to have its parameters adjusted so it will be able to reflect the behaviour of a given operator. Thus, these Membership Functions (MFs) have to be appropriately adjusted so that the generated control signal approximates the HO behaviour.

One way to cope with this disadvantage of the model-based FL approach is to come up with an automatic procedure for finding the appropriate adjustments of the MFs. In this work, this procedure is called "tuning". As there are many possible combinations for the MFs parameters, the search space for the tuning procedure is inevitably large. To solve such kind of high dimensional search space problems, Genetic Algorithms can be applied (Orvosh and Davis, 1994). In this work, a Genetic Algorithm (GA) is developed to tune and validate the proposed FL model.

The remainder of this work is organised as follows. Section 2 gives a general idea of the desired behaviour of the developed controller based on a hypothetic model for the way the HO acts in a manual control loop. In Section 3, the controller is

formally presented and its natural extension, via FL approach is achieved. In Section 4, a Genetic Algorithm (GA) is used to select the appropriate FLC parameters. A nonlinear Process Control application is tested with the developed FLC to compare the generated control actions with the manual operation in Section 5, where the results of the system simulation and discussion are presented. Section 6 summarises the conclusions of this paper and recommendations for future work.

2. HUMAN OPERATOR BASED INTUITIVE CONTROLLER DEVELOPMENT

2.1 Human Operator tasks and responsibilities

In a process plant, commonly, the HO has the responsibility of maintaining the plant under control, mainly by manipulating the final control elements, in manual loops, or by changing the controllers' set-point (SP) values.

The first concern of the operators is about safety. Right after the security concern is the production task. The production throughput should not decrease in time. Supervisors are always checking for production problems and possible causes of such incidents.

2.2 Human Operator's behaviour model

Two modes of operation may be defined for the HO:

A. When changing the operating conditions (SP-Tracking);

B. When rejecting disturbances (Disturbance Rejection).

In the first mode of operation (Mode A), the operator, to not disturb the system, will change the operating conditions only when necessary by slow changes in the final control element. This tends to minimise some problems such as interactions between loops, for example. This manual procedure is equivalent to changing the controller SP, when it is in automatic. Hence, Mode A is called SP-Tracking mode of operation. In Mode B, to reject a given disturbance, the behaviour of the operator is normally more aggressive. This is natural, since his task is to maintain the process plant under control.

An intuitive algorithm to describe the way the operator adjusts the final control element (Control Valve, for example), considering a single input single output system, subject to disturbances, can be described as follows:

1) Is the PV following the desired path (SP)?

If 'Yes' then "Do nothing. The process is under control"

Else

If Mode A: - apply Mode A procedure;

If Mode B: - apply Mode B procedure;

End

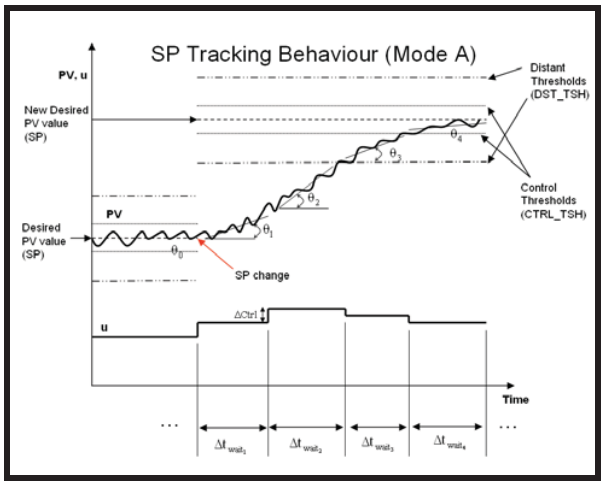


Fig.2: Intuitive HO behaviour when in Mode A of operation.

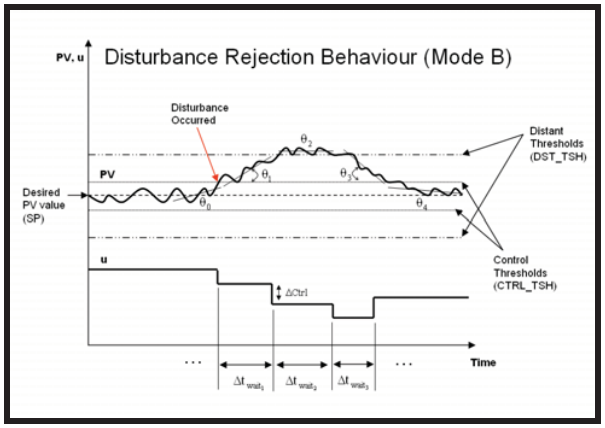


Fig. 3: Intuitive HO behaviour when in Mode B of operation.

2) When in Mode A:

Manipulate the Control Valve appropriately and wait for the system to react. If the trend of the PV is already going to the desired SP with an appropriate "velocity" do not change the Control Valve value. However, if the PV trend is going too "fast" or too "slow" to the desired SP, change the Control Valve appropriatelly and wait for the system's response

3) When in Mode B:

Perform the same actions done in Mode A, but with more aggressiveness, that depends upon the value of the PV.

From Fig. 2 and Fig. 3, some subjective terms mentioned in items 1, 2 and 3 such as "velocity", "slow" and "fast", are clarified. It can be observed that as the operator inspects the PV, he determines if the PV is under control by observing three variables, mainly:

- Variable 1 – Angle that the PV trend makes with respect to the desired SP;
- Variable 2 – Distance between the PV and SP;
- Variable 3 – Is the error increasing or decreasing?

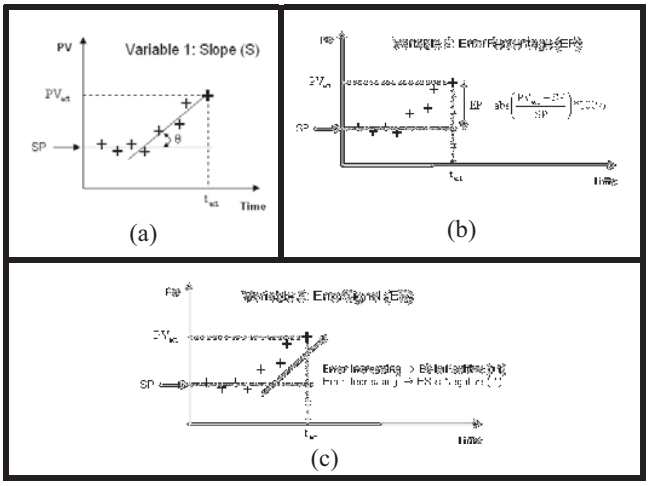


Fig.4: Variables used to encapsulate the HO's behaviour in the HOBIC.

Hence, the action taken in the Control Valve will be generated after analysing these three variables. After the action, the operator has to wait some time until the system reacts to it. The minimum time to wait (Δt_{wait}) should be greater than the Process time-delay. Thus, after observing the result of his action, the HO judges again the variables 1, 2 and 3 and takes another action or waits, if the PV is already under control again or if the PV trend is behaving as intended.

The PV is considered to be "behaving as intended" if it is approaching the SP within a given range of angles ("velocity") at a given distance from the SP that the operator establishes in his mind for that specific system. Therefore, if the angle is not within the desired range of values for a specific distance away from SP, then the control action is increased or reduced appropriately. From Fig.2 and Fig.3 it is clear that for SP-tracking (Mode A) the actions are less aggressive than when rejecting disturbances (Mode B). These figures also show that the HO has in his mind imaginary thresholds to determine how far from the SP the PV is (CTRL_TSH and DST_TSH).

2.3 The HOBIC and its natural extension –Fuzzy-HOBIC

Fig. 4 suggests the way variables 1, 2 and 3 are obtained. Variable 3, denoted as ErrorSignal (ES), reflects whether the error between SP and PV is increasing ("+1") or decreasing ("-1"). Variable 2, shown in Fig. 4(b), defines the absolute value of the error percentage between PV_{act} and SP, i.e. ErrorPercentage (EP). The reason for defining the distance between PV and SP as an error percentage measure is because the operator tends to analyse the PV value relatively to its desired value to judge if the PV is close or far from the SP. For example, for SP values of 100 units, deviations of 3 units can be considered to be "small" (3%) by the operator, and no action would be taken. However, if the SP is zero, the EP will be, by definition, the absolute value of the PV times 100%. Variable 1 defines the angle, in degrees, that the PV trend makes with the SP, denoted by Slope (S) in Fig.4(a).

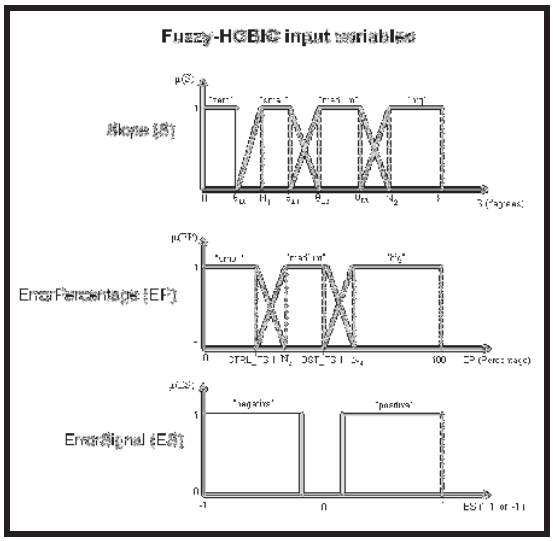


Fig. 5: Fuzzy-HOBIC Linguistic variables.

The Slope can be obtained numerically using Multi-variable Least Squares (LS). From Fig. 4(a), one can notice that the angle is obtained by using five samples (PV_{act} and the past four samples). This is a good compromise between being less sensitive to the presence of noise and getting the actual PV trend. It is being assumed here that the sampling period used is sufficiently low to capture the system dynamics (eg.: 10% of the dominant time-constant) and sufficiently high to not to capture only the noise dynamics.

After defining how the variables 1, 2 and 3 are determined in the HOBIC, the next step is to specify the thresholds CTRL_TSH and DST_TSH. The Control Threshold is, by definition, the limits within which the operator judges that the system is under control and no action is taken, when the PV has "small" Slope values. The Distant Threshold is obtained by determining the distance between PV and SP, when the operator's actions start to increase significantly. These limits are also automatically detected (Section 3).

2.4 Determining the HOBIC's rules

To be able to embed in the HOBIC the rules that the operator is using, four angle limits are defined: $\theta_{L0}, \theta_{L1}, \theta_{L2}, \theta_{L3}$. The first angle limit (θ_{L0}) is a dead band limit. In other words, the HOBIC will consider that the Slope is zero if S is less than θ_{L0}. The other three limits are distributed from θ_{L0} to 90 degrees, dividing this region into intervals. For each region of Slope values and considering the EP and ES variables, a specific action is taken in the Control Valve. Judgment about what action is to be taken given the system state (S, EP, ES) is performed by a set of 20 rules. However, these rules can be simplified by applying the FL approach.

The actual HOBIC variables S, EP and ES are considered to be linguistic input variables. The Linguistic values for these variables are as follows:

Table 1: Fuzzy-HOBIC rules definition.

Rule N°	Rule Definition	*Abs (deltaAction)*
1	IF (S is zero) and (EP is small)	*zero*
2	IF (S is zero) and (EP is medium)	*small*
3	IF (S is zero) and (EP is big)	*medium*
4	IF (S is small) and (EP is small)	*small*
5	IF (S is small) and (EP is medium)	*small*
6	IF (S is small) and (EP is big)	*medium*
7	IF (S is medium) and (EP is small)	*small*
8	IF (S is medium) and (EP is medium) and (ES is negative)	*Zero*
9	IF (S is medium) and (EP is medium) and (ES is positive)	*medium*
10	IF (S is medium) and (EP is big) and (ES is negative)	*zero*
11	IF (S is medium) and (EP is big) and (ES is positive)	*big*
12	IF (S is big) and (EP is small)	*medium*
13	IF (S is big) and (EP is medium)	*medium*
14	IF (S is big) and (EP is big) and (ES is negative)	*small*
15	IF (S is big) and (EP is big) and (ES is positive)	*max*

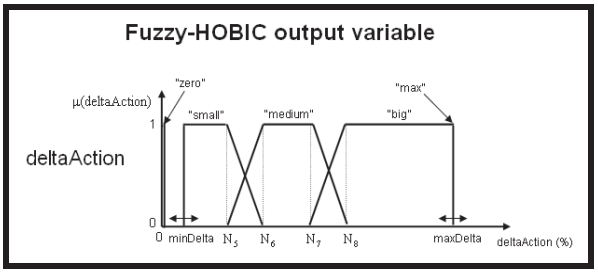


Fig. 6: Fuzzy-HOBIC output variable description.

• Slope (S): “zero”, “small”, “medium”, “big”

• ErrorPercentage (EP): “small”, “medium”, “big”

• ErrorSignal (ES): “positive”, “negative”

The Fuzzy-HOBIC input variables are described in Fig. 5. Each linguistic value is mathematically defined as a MF. Applying FL approach, the rules number is reduced to 15. They are shown in Table 1. This happens without loss of generality because of the advantage that the fuzzy rules give of activating more than one rule at a time.

It is important to notice that the control action is shown in Table 1 as an absolute value. The sign of deltaAction, is determined by observing the ES value. The Fuzzy-HOBIC output variable (deltaAction) is shown in Fig.6. About Fig. 6, the linguistic values “zero” and “max” are applied to force the Defuzzification process to give the numeric outputs zero and maxDelta, according to its respective fuzzy rules.

The process time-delay, minDelta and maxDelta values are assumed to be known inputs that depend upon the application and the HO’s behaviour, as well as the times involved to wait for the system to react, after the control actions are given.To cope with the disadvantage of having many parameters to tune for this controller, an automatic method of tuning the developed Fuzzy-HOBIC using a Genetic Algorithm (GA) is developed.

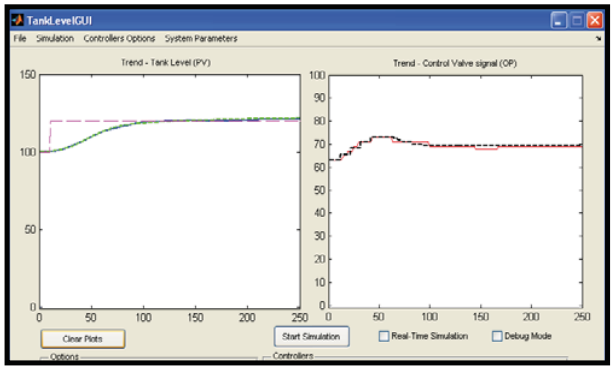


Fig. 7: Fuzzy-HOBIC (dashed lines) vs. Manual Operation (solid lines). Step up test (+20%).

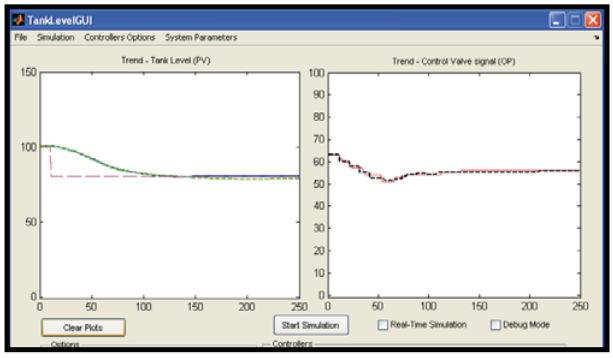


Fig. 8: Fuzzy-HOBIC (dashed lines) vs. Manual Operation (solid lines). Step down test (-20%).

3. APPLYING A GENETIC ALGORITHM TO TUNE THE FUZZY-HOBIC

The objective of the GA is to find values of the 14 parameters (P1-P14) that will make the Fuzzy-HOBIC approximate a given HO’s behaviour. The closer the Fuzzy-HOBIC’s actions are to the HO’s ones the better is the tuning. A suitable objective function, is given by (1), where $U_{man}(i)$ and $U_{FHOBIC}(i)$ represent the sample ‘i’ of the manual and the Fuzzy-HOBIC actions from a total of N

available samples, respectively.

$$J = \sum_{i=1}^{N} \frac{(U_{man}(i) - U_{FHOBIC}(i))^2}{N} \quad (1)$$

For the developed GA, an elitist strategy is used (Chipperfield, Fleming, Pohlheim and Fonseca, 1994). The initial population is split into two sets which are used to compose three sub-populations. The first set is composed of the best individuals of the population (smallest J values). This set composes the first and the second sub-populations. The first one has a low mutation rate, while in the second a high mutation rate is applied. The low mutation rate in the first sub-population is used to search for local minimums, while the high mutation rates for the second population is applied to find new regions of minimums, trying to avoid getting trapped in local minimums.

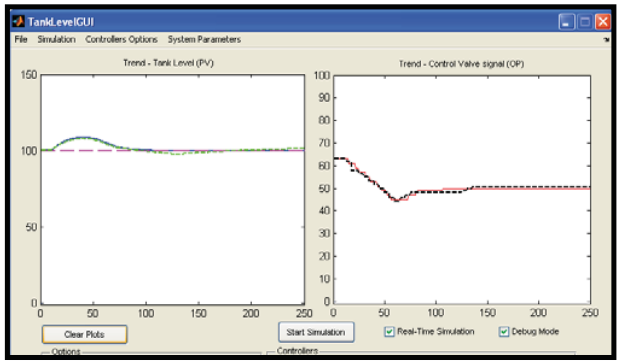


Fig. 9: Fuzzy-HOBIC (dashed lines) vs. Manual Operation (solid lines). Disturbance Rejection up test (+20%).

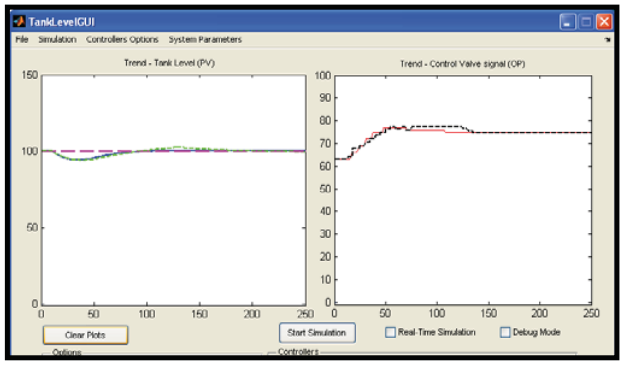


Fig. 10: Fuzzy-HOBIC (dashed lines) vs. Manual Operation (solid lines). Disturbance Rejection down test (-20%).

A third sub-population is composed of the second set of the population. In this case, a high mutation is applied, because of the same reasoning used for the second population.

The convergence criteria applied in this work is either when the best individual from the population does not change for more than 10 generations or when the maximum number of generations is exceeded.

4. RESULTS AND DISCUSSION

The application chosen for testing the Fuzzy-HOBIC is controlling the level of liquid in a Tank, in the same manner as performed by Enab (2005). This is a very common nonlinear application in the process industry. For this specific application, it is also supposed that the level needs a tight control. The input flow control valve is used to regulate the tank level, while the output flow control valve generates the non-measured disturbance. After defining the application to test the Fuzzy-HOBIC it is necessary to develop a simulation environment that reflects the proposed system to be controlled. A Graphical User Interface (GUI) was implemented to simulate the tank level system.

Figures 7-10 show the results obtained when tuning the Fuzzy-HOBIC using the GA approach. It is important to notice, however, that two different tunings where used here: one for sp-tracking and the other for disturbance rejection. The manual operations where generated using the developed GUI, by an operator that got experienced by using the system. When controlling the level in manual, two objectives where followed: 1) Do not produce any overshoot, when tracking set-point; 2) Try to eliminate the disturbance as fast as possible without making large changes in the control valve. These objectives are in accordance with the HO's behaviour, described in sub-section 2.2. As can be seen by the results, the operator's behaviour could be well approximated the tuned Fuzzy-HOBICs, showing its "stair-like" signals nature.

5. CONCLUSIONS

The results of applying the Fuzzy-HOBIC in a process control simulation have indicated that:

• The rules used to describe the HO's behaviour were adequate for approximating his manual operations in the application tested;

• A process model is not needed to tune the Fuzzy-HOBIC.

As recommendations for future work, it is suggested to test the developed controller in a real Process Control Application. Another possible application of Fuzzy-HOBIC would be to train apprentice operators, as already suggested by Zapata, Galvão, and Yoneyama (1999).

REFERENCES

Bialkowski, W. L. (1993). Dream versus Reality: a view from both sides of the gap, Pulp and Paper Canada, 94 (11), pp. 19-27.

Costello, R. G. and Higgins, T. J. (1966). An Inclusive Classified bibliography pertaining to modelling the human operator as an element in an automatic control system. *VII IEEE Symp. On Human Factors in Electronics*, Minneapolis, Minn.

Chipperfield, A., Fleming, P., Pohlheim, H. and Fonseca, C. (1994). Genetic Algorithm Toolbox: For use with MATLAB - Software and User's Guide. Department of Automatic Control and Systems Engineering. University of Sheffield. UK.

Desborough, L. and Miller, R. (2001). Increasing Customer Value of Industrial Performance Monitoring – Honeywell's Experience. Honeywell Hi-Spec Solutions.

Enab, Y. M. (1995). Controller Design for an unknown process, Using simulation of a Human Operator, *Engineering Appl. of Artificial Intelligence* 8(3), pp. 299-308.

Ertugrul, S. E. and Hizal, N. A. (2005). Neuro-Fuzzy controller design via modelling human operator actions. *Journal of Intelligent & Fuzzy Systems* 16, 133-140, IOS Press.

Kleinman, D. L., Baron, S. and Levison, W. H. (1970). An Optimal Control Model of Human Response – Part I: Theory and Validation. *Automatica*, vol. 6, pp. 357-369.

Orvosh, D. and Davis, L. (1994). Using a Genetic Algorithm to Optimize Problems with Feasibility Constraints. *IEEE Conference on Evolutionary Computation - Proceedings* (2/-), pp. 548-553.

Zapata, G. O. A., Galvão, R. K. H. and Yoneyama, T. (1999). Extracting Fuzzy Control Rules from Experimental Human Operator data. *IEEE Trans. On Systems, Man and Cybernetics-PartB: Cybernetics* 29(3), pp. 398-406.

Considerations on Set-Point Weight choice for 2-DoF PID Controllers

Víctor M. Alfaro [*] **Ramon Vilanova** [**] **Orlando Arrieta** [*,**]

[*] *Departamento de Automática, Escuela de Ingeniería Eléctrica Universidad de Costa Rica, San José, 11501-2060 Costa Rica. e-mail: {Victor.Alfaro, Orlando.Arrieta}@ucr.ac.cr*
[**] *Departament de Telecomunicació i d'Enginyeria de Sistemes, ETSE Universitat Autònoma de Barcelona, 08193 Bellaterra, Barcelona, Spain. e-mail: {Ramon.Vilanova, Orlando.Arrieta}@uab.cat*

Abstract: This paper's aim is to present an analysis of the influence of the 2-DoF controllers proportional set-point weight over the servo-control performance and to show that the removal of the existing constraint for its selection ($0 \leq \beta \leq 1.0$) will allow to improve its performance when a high robust regulatory control system is required. A concrete analysis is conducted by using 2-DoF PID tuning approaches that explicitly take the desired robustness level as a design parameter. It is seen that as the desired robustness increases the tuning methods suggest values $\beta > 1.0$. Performance looses are evaluated if we are to be constrained to the case $\beta \leq 1.0$.

Keywords: PID Control, Two-Degree-of-Freedom, Set-Point Weights

1. INTRODUCTION

Since their introduction in 1940 (Babb, 1990; Bennett, 2000) commercial *Proportional - Integrative - Derivative* (PID) controllers have been with no doubt the most extensive option found on industrial control applications. Their success is mainly due to its simple structure and meaning of the corresponding three parameters. This fact makes PID control easier to understand by the control engineers than other most advanced control techniques.

With regard to the design and tuning of PID controllers, there are many methods that can be found in the literature over the last sixty years. Special attention is made of the IFAC workshop PID'00 Past, Present and Future of PID Control held in Terrassa, Spain, on April 2000, where a glimpse of the state-of-the-art on PID control was provided. It can be seen that most of them are concerned with feedback controllers which are tuned either with a view to the rejection of disturbances (Cohen and Coon, 1953; López et al., 1967; Ziegler and Nichols, 1942) or for a well-damped fast response to a step change in the controller set-point (Martin et al., 1975; Rivera et al., 1986; Rovira et al., 1969). The Two-Degree-of-Freedom (2-DoF) formulation is aimed at trying to met both objectives. This second degree of freedom is aimed at providing additional flexibility to the control system design. See for example (Araki, 1984a,b, 1985) and its characteristics revised and summarized in (Taguchi and Araki, 2000, 2002) and (Taguchi et al., 2002), as well as different tuning methods that have been formulated over the last years (Alfaro et al., 2008; Åström et al., 1992; Åström and Hägglund, 2004; Åström et al., 1998; Gorez, 2003; Hang and Cao, 1996; Hägglund and Åström, 2002; Taguchi and Araki, 2000).

This second degree of freedom is found on the presented literature as well as in commercial PID controllers under the form of the well known set-point weighting factor (usually called β) that ranges within $0 \leq \beta \leq 1.0$, being the main purpose of this parameter to avoid excessive proportional control action when a set-point change takes place. Therefore the use of *just a fraction* of the set-point.

There is however a shift of perspective with the introduction of Robustness considerations (Åström and Hägglund (1995, 2004, 2006)). As a result, less aggressive control actions are generated and smooth responses are achieved. However, if the desired level of robustness is high, step response performance can be seriously degraded. This is the analysis conducted in this paper that leads us to conclude that the use of values of β that are beyond the constraint, are definitively needed in order to get better step response performance. The analysis is conducted by using two existing tuning rules for 2-DoF PID controllers that include, as an explicit design parameter, the desired robustness level in terms of the Maximum Sensitivity value. This allows to analyze the effect of going on increasing the desired robustness level. The suggested value for β goes to values $\beta > 1.0$ in many of the cases therefore constraining the achievable performance if we are to be limited to a maximum of $\beta = 1.0$.

It is worth to notice that even the suggestion of allowing $\beta > 1.0$ seems quite common sense and natural, to the knowledge of the authors it has not still been considered. In this paper this proposal is raised within the motivation of the increased use of robustness considerations on what we could call *modern* control design approaches.

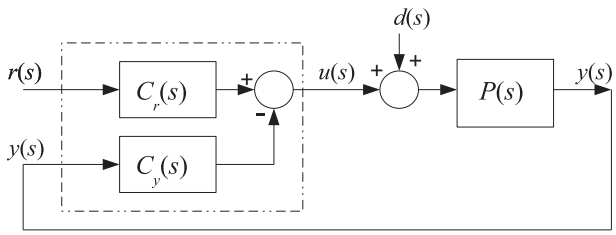


Figure 1. Closed-loop Control System

The paper is organized as follows. Section 2 introduces the control setup and the 2-DoF PID formulation. Discussion about the selection of the β parameter is also introduced. Section 3 analyzes the effects of constraining the set-point weight on robustness based tuning rules and suggests to relax that constraint in order to met a high demanding robustness-performance *tradeoff*. On Section 4 an example illustrates how performance increases if larger values of β are allowed. The paper ends in Section 5 with some conclusions.

2. 2-DOF PID FORMULATION

Consider the closed-loop control system of Fig. 1, where $P(s)$ is the *controlled process* transfer function, $C_r(s)$ the *set-point controller* transfer function, $C_y(s)$ the *feedback controller* transfer function, and $r(s)$ the *set-point*, $d(s)$ the *load-disturbance*, and $y(s)$ the *controlled variable* (process output).

The output of the controller is given by

$$u(s) = C_r(s)r(s) - C_y(s)y(s) \tag{1}$$

Without loss of generality we will use an error feedback *Ideal* PID controller which equation is

$$\begin{aligned} u(s) = & K_c \left(1 + \frac{1}{T_i s}\right) r(s) \\ & - K_c \left(1 + \frac{1}{T_i s} + \frac{T_d s}{T_d/N s + 1}\right) y(s) \end{aligned} \tag{2}$$

where K_c is the *controller gain*, T_i the *integral time constant*, T_d the *derivative time constant* and N the *derivative filter constant* (usually $N = 10$ (Visioli, 2006)). Then, the controllers' transfer functions are

$$C_r(s) = K_c \left(1 + \frac{1}{T_i s}\right) \tag{3}$$

and

$$C_y(s) = K_c \left(1 + \frac{1}{T_i s} + \frac{T_d s}{0.1 T_d s + 1}\right) \tag{4}$$

The closed-loop control system response to a change in any of its inputs, will be given by

$$y(s) = \frac{C_r(s)P(s)}{1 + C_y(s)P(s)} r(s) + \frac{P(s)}{1 + C_y(s)P(s)} d(s) \tag{5}$$

or in a compact form by

$$y(s) = M_{yr}(s)r(s) + M_{yd}(s)d(s) \tag{6}$$

where $M_{yr}(s)$ is the transfer function from set-point to controlled process variable: the *servo-control* closed-loop transfer function or complementary sensitivity function; $T(s)$; and $M_{yd}(s)$ is the one from load-disturbance to controlled process variable: the *regulatory control* closed-loop transfer function or disturbance sensitivity function $S(s)$.

Since all parameters of $C_r(s)$ are identical to the ones of $C_y(s)$ it is not possible to specify the dynamic performance of the control system to set-point changes, independently of the performance to load-disturbances changes.

If the degrees of freedom in control system are defined as the number of closed-loop transfer functions that may be selected independently (Horowitz, 1963), we have in this case a *One-Degree-of-Freedom* (1-DoF) control system.

The above constraint forces the designer to use a tuning rule developed for the specific required application (servo-control o regulatory control) finding in the literature tuning rules for regulatory control (Cohen and Coon, 1953; López et al., 1967; Ziegler and Nichols, 1942), for servo-control applications (Martin et al., 1975; Rivera et al., 1986; Rovira et al., 1969) or separate tuning rules for both applications (Chien et al., 1952; Kaya, 2004; Sung and Lee, 1999) only to mention a few. Alternatively the 1-DoF can be forced to operate in order to provide a balanced performance with respect to both operation modes. This is the so called *implicit* 2-DoF PID and has been presented in (Arrieta and Vilanova (2007b,c); Arrieta et al. (2008)). A collection of tuning methods may be found in O'Dwyer (2003).

It has been widely reported elsewhere that a control system with a controller optimized for load-disturbance rejection, normally presents high overshoots to set-point step inputs requiring a detuning with the consequential reduction in its regulatory performance. In such case, considerations about performance degradation of optimal tunings have to be taken into account (Arrieta and Vilanova (2007a)).

In order to provide additional flexibility for the control system design, a second degree of freedom was introduced into the PID algorithms in Araki (1984a,b, 1985) and its characteristics revised and summarized in Taguchi and Araki (2000, 2002) and Taguchi et al. (2002).

Consider now the PID controller equation (Åström and Hägglund, 2006)

$$\begin{aligned} u(s) = & K_c \left(\beta + \frac{1}{T_i s} + \frac{\gamma T_d s}{0.1 T_d s + 1}\right) r(s) \\ & - K_c \left(1 + \frac{1}{T_i s} + \frac{T_d s}{0.1 T_s s + 1}\right) y(s) \end{aligned} \tag{7}$$

where β and γ are the *set-point weights*.

The γ parameter is more frequently applied as a derivative mode *switch* (0 or 1) for the signal reference r. To avoid extreme instantaneous change in the controller output signal when a set-point step change occurs normally γ is set to zero. In this case the new set-point controller transfer function is

$$C_r(s) = K_c \left(\beta + \frac{1}{T_i s}\right) \tag{8}$$

and the one for the feedback controller

$$C_y(s) = K_c \left(1 + \frac{1}{T_i s} + \frac{T_d s}{0.1 T_d s + 1}\right) \tag{9}$$

which is the same as (4) above.

In commercial controllers the proportional set-point weight β may be selected only in the $0 \leq \beta \leq 1.0$ range.

Given a controlled process $P(s)$, the feedback controller $C_y(s)$ parameters (K_c, T_i, T_d) may be selected to achieve a target performance for the regulatory control $M_{yd}(s)$,

and then using the proportional set-point weight (β), in the set-point controller $C_r(s)$, to modify the servo-control performance $M_{yr}(s)$.
Under the above degree of freedom definition, we have now a *Two-Degree-of-Freedom* (2-DoF) control system. This option allowed the development of sets of tuning methods for the 2-DoF controllers as the ones found in (Alfaro et al., 2008; Åström et al., 1992; Åström and Hägglund, 2004; Åström et al., 1998; Gorez, 2003; Hang and Cao, 1996; Hägglund and Åström, 2002; Taguchi and Araki, 2000).
With regard to the commercial implementation of the PID algorithms, it is usually to find that most of them are of 1-DoF type like the ones described in (ABB ((n.d.); Foxboro (1998); Fuji (2001); Honeywell (2007); Rockwell (2003, 2005)) a few include a set-point filter ((Omron, 2007; Yokogawa, (n.d.)) and very few have 2-DoF capabilities ((Emerson, 2008; Mitsubishi, 2002)). In particular, the 2-DoF PID controller in (Emerson (2008)) allows both weights in (7) (β and γ) to be selected in the full 0 to 1 range.

3. PROPORTIONAL SET-POINT WEIGHTING ANALYSIS

From (5) and (6) the servo-control closed-loop transfer function is

$$M_{yr}(s) = \frac{C_r(s)P(s)}{1 + C_y(s)P(s)} \tag{10}$$

and the one for the regulatory control

$$M_{yd}(s) = \frac{P(s)}{1 + C_y(s)P(s)} \tag{11}$$

which are related by

$$M_{yd}(s) = C_r(s)M_{yr}(s) \tag{12}$$

Using (8) in (12) we have

$$M_{yd}(s) = K_c \left(\frac{\beta T_i s + 1}{T_i s} \right) M_{yr}(s) \tag{13}$$

On the other hand, the characteristic polynomial of the closed-loop control system is

$$p(s) = 1 + C_y(s)P(s) \tag{14}$$

from where it can be obtained the closed-loop poles location; therefore the closed-loop stability; depends only on the $C_y(s)$ parameters, hence not affected by β.
This fact makes possible to design first the feedback controller considering the *regulatory control performance* and the *closed-loop control system robustness* and, on a second step to modify the set-point controller considering only the *servo-control performance* (by the introduction of β).
Although, the instant change in the controller output signal to a step set-point change is given by

$$\Delta y_r = K_c \beta \Delta e_r = K_c \beta \Delta r \tag{15}$$

Since the performance optimization of a regulatory control system requires controllers' gains higher than the optimization of the same loop for servo-control operation, the use of a proportional set-point weight $\beta < 1$ allows to shift to the left the controller integral mode zero to a desired position to reduce the controlled signal overshoot and also to decrease the instant change in the controller output.
From the above presented analysis, it is clear that the use of a 2-DoF controller improves the servo-control performance and no questions arise about the manufactures imposed constraint on the proportional set-point weight selection range. It is along this framework that the above indicated tuning rules for 2-DoF PI and PID controllers; including the ones that take into consideration the control system robustness; constraint the set-point weight to $0 \leq \beta \leq 1$ respecting the allowed range in commercial controllers (see for example (Alfaro et al., 2008; Åström and Hägglund, 2004; Åström et al., 1998; Gorez, 2003; Hägglund and Åström, 2002; Taguchi and Araki, 2000)).
However, within a more *modern* framework robustness considerations are an integral part of practically every design approach. In such cases, as it will be explicitly shown in next section, extending the allowed range for the set-point weight will be definitively needed in order to be able to improve the servo-control performance. When a highly robust control system is required due to the expected variations in the controlled process characteristics, a significant reduction in the controller gain is needed and the performance of the control-loop will decrease. The system responses to load-disturbance and set-point changes will be slower.
This situation motivates the analysis of the proportional set-point weighting effect over the control system performance when the set-point changes, using two of the available tuning rules for 2-DoF controllers. The choice of the presented tuning rules is based on the fact that they include, as an explicit design parameter, the desired robustness level for the closed-loop control system. This setup allows for a more concrete and objective analysis. However the analysis can be easily extended to other tuning rules as the effect of getting a more robust feedback system is by sure to generate more conservative responses.

3.1 ART_2 PI Controller Tuning

The *Analytical Robust Tuning* for 2-DoF PI controllers (ART_2) follows (Alfaro et al., 2008) and is outlined here:

- *Controlled Process Model:*
$$P(s) = \frac{K_p e^{-Ls}}{Ts + 1} \tag{16}$$
where K_p is the *process gain*, T is the *time constant*, and L is the *dead-time*. It will be referred to $\tau_o = L/T \leq 1.0$ as the controlled process *normalized dead-time*.
- *Controller's Parameters:* The ART_2 tuning equations are
$$\kappa_c = K_c K_p = \frac{2\tau_c - \tau_c^2 + \tau_o}{(\tau_c + \tau_o)^2} \tag{17}$$
$$\tau_i = \frac{T_i}{T} = \frac{2\tau_c - \tau_c^2 + \tau_o}{1 + \tau_o} \tag{18}$$
where κ_c and τ_i are the controller *normalized parameters* and $\tau_c = T_c/T$ the *design parameter* (T_c is the target regulatory control closed-loop time constant).
- *Set-point Weighting:* The proportional set-point weight selection criteria is
$$\beta = \min \left\{ \frac{1}{K_c}, \frac{\tau_c T}{T_i}, 1 \right\} \tag{19}$$
- *Design Parameter:* The design parameter τ_c may be selected within the range
$$\max(0.50, \tau_{cmin}) \leq \tau_c \leq 1.50 + 0.3\tau_o \tag{20}$$
where τ_{cmin} is given by

$$\tau_{cmin} = k_{11}(M_s) + \left[\frac{k_{21}(M_s)}{k_{22}(M_s)}\right]\tau_o \qquad (21)$$

$$k_{11}(M_s) = 1.384 - 1.063M_s + 0.262M_s^2$$

$$k_{21}(M_s) = -1.915 + 1.415M_s - 0.077M_s^2$$

$$k_{22}(M_s) = 4.382 - 7.396M_s + 3.0M_s^2$$

allowing to design the control system with a robustness higher than the minimum required (give it by the maximum sensitivity M_s).

Using (20) and (21), the lower limits for the design parameter τ_c may be estimated. These are shown in Table 1 for robustness $1.2 \leq M_s \leq 2.0$ and controlled process model normalized dead-time $0.1 \leq \tau_o \leq 1.0$.
As it can be seen in Table 1 the lower and higher recommended limits in (20) were reached for the extreme cases (low normalized dead-time and robustness and high normalized dead-time and robustness). For the first case this means that, slow responses with high robustness system requirements will be obtained, and for the second one, that it is not possible to obtain a system with the high robustness specified.
The controller's proportional set-point weight may be obtained with (19) and they are shown in Table 2. As can be seen in this Table the existing upper limit constraint of 1.0 for β was intentionally relaxed (bold).

According to the ART_2 tuning rules, when the normalized dead-time is in the upper side of the range ($\tau_o \approx 1$) and the required system robustness is high, the recommended proportional weight would be higher than 1.0. As it can be seen in the last column of Table 2, this is the situation for practically all values of τ_o when a robustness $M_s = 1.2$ is specified. Therefore, the imposed constraint for the β value selection, in the available commercial Two-Degree-of-Freedom PID controllers, does not allow the designer to use the full capabilities of these controllers.

Table 1. Higher Close-loop Speed Allowed τ_{cmin}

	M_s				
τ_o	2.0	1.8	1.6	1.4	1.2
0.1	0.500	0.500	0.500	0.501	0.675
0.2	0.500	0.500	0.500	0.593	0.864
0.3	0.500	0.500	0.553	0.685	1.054
0.4	0.500	0.513	0.620	0.777	1.243
0.5	0.500	0.562	0.686	0.869	1.432
0.6	0.535	0.610	0.753	0.961	1.622
0.7	0.573	0.659	0.819	1.053	1.710
0.8	0.611	0.707	0.886	1.145	1.740
0.9	0.650	0.756	0.952	1.236	1.770
1.0	0.688	0.804	1.019	1.328	1.800

Table 2. Proportional Set-Point Weight Factor β

	M_s				
τ_o	2.0	1.8	1.6	1.4	1.2
0.1	0.424	0.424	0.424	0.425	0.604
0.2	0.516	0.516	0.516	0.608	0.878
0.3	0.609	0.609	0.654	0.742	**1.056**
0.4	0.609	0.618	0.691	0.806	**1.298**
0.5	0.600	0.644	0.735	0.879	**1.636**
0.6	0.619	0.674	0.783	0.962	**2.138**
0.7	0.642	0.707	0.835	**1.054**	**2.431**
0.8	0.667	0.743	0.892	**1.158**	**2.501**
0.9	0.694	0.780	0.954	**1.274**	**2.573**
1.0	0.723	0.820	**1.019**	**1.404**	**2.647**

3.2 Integrated Absolute Error (IAE) Optimized PID Tuning

A tuning method for 2-DoF PI and PID controllers that optimizes their performance under a IAE cost functional ensuring at the same time a minimum closed-loop robustness (M_s) is described in Méndez (2008). Bellow are presented the controller parameters for one particular robustness level, say $M_s = 1.4$.

- *Controlled Process Model:*
$$P(s) = \frac{K_p e^{-Ls}}{(Ts+1)(aTs+1)} \qquad (22)$$
where K_p is the process gain, T is the dominant time constant, a is the *time constants ratio* and L is the dead-time ($\tau_o = L/T$).
- *Controller's Set-Point Weight:* Table 3 shows the PID_2 controller's set-point weighting factor for the $M_s = 1.4$ case corresponding to different values of τ_o and a. As shown in this table for this robustness most of the recommended proportional set-point weights exceed the 1.0 upper limit (bold).

Table 3. $IAE - M_s$ PID_2 Controller Set-Point Weight β

	τ_o						
a	0.1	0.25	0.50	0.75	1.0	1.50	2.0
0.25	0.636	0.819	**1.073**	**1.256**	**1.413**	**1.665**	**1.839**
0.50	0.585	0.731	0.992	**1.189**	**1.338**	**1.630**	**1.778**
0.75	0.588	0.695	0.921	**1.104**	**1.261**	**1.550**	**1.755**
1.0	0.567	0.662	0.871	**1.052**	**1.210**	**1.412**	**1.663**

4. EXAMPLE

This section provides an example to show the effect of the proportional set-point weighting over the servo-control system performance.
In order to have simulation results more close to industrial practice, in the example it is assumed that all variables can vary in the 0 to 100% normalized range and that in the normal operation point, the controlled variable, the set-point and the control signal, have all values close to 70%. For the tests a 20% change in set-point followed by a 10% change in load-disturbance will be used in all cases.
Performance: Performance will be evaluated for a set-point change and under the presence of a load-disturbance. The Integrated-Absolute-Error (IAE) that is defined as

$$J_{IAE} \doteq \int_0^\infty |r(t) - y(t)|\, dt \qquad (23)$$

and provides a measure for control system *output performance*.
Control input usage: On the other hand to evaluate the manipulated input usage, the total variation of the control effort $u(t)$ (TV_u) is computed. This value is defined, for a discrete signal as the sum of the size of its increments

$$TV_u \doteq \sum_{k=1}^{\infty} |u_{k+1} - u_k| \qquad (24)$$

This quantity should be as small as possible and provides a measure of the *smoothness of the control signal*.
Robustness: The maximum sensitivity value

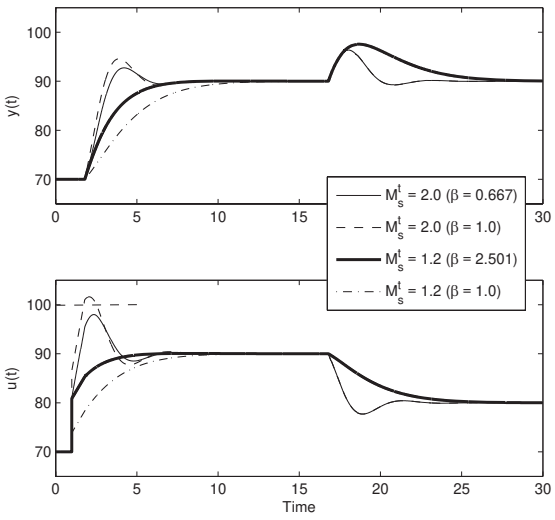


Figure 2. PI Control System Responses

Table 4. ART_2 PI Controller Parameters

M_s^t	τ_c	K_c	T_i	β
2.0	0.611	0.828	0.916	0.667
1.2	1.740	0.194	0.988	2.501

$$M_s = \max_{\omega} |S(\mathrm{j}\omega)| = \max_{\omega} \frac{1}{|1 + C_y(\mathrm{j}\omega)P(\mathrm{j}\omega)|} \qquad (25)$$

is used as a measure of the control system *robustness*. Recommended values for M_s are typically within the range 1.2 - 2.0.

Consider the particular case of controlled process (16) with $K_p = 1.0$, $T = 1.0$ and $L = 0.80$ ($\tau_o = 0.80$). The PI_2 controller's parameters obtained with the ART_2 tuning method in Section 3.1, in order to have a low robustness ($M_s^t = 2.0$) and a high robustness ($M_s^t = 1.2$) control system, are shown in Table 4 (the upper limit constraint for β was not taken into account).

System responses are shown in Fig. 2. The figure includes also the responses obtained with $\beta = 1.0$ in both cases.

The servo-control performance (J_{IAEr}) and control effort smoothness (TV_{ur}) as well as the obtained control system robustness (M_s^r) are shown in Table 5.

For the low robustness case ($M_s^t = 2.0$) the use of a proportional set-point weight lower than 1.0 ($\beta = 0.667$) allows to reduce: the servo-control controlled variable overshoot, the control effort upper value and helps to made it smoother compared with the $\beta = 1.0$ case. This last case is equivalent to the use of a 1-DoF PI controller.

In the high robustness case ($M_s^t = 1.2$) the use of a proportional set-point weight higher than 1.0 ($\beta = 2.501$) allows to improve the servo-control performance reducing J_{IAEr} without deterioration of the control effort behavior TV_{ur} compared with the case of $\beta = 1.0$. This is when the use of the set-point weight, in the 2-DoF PI controller, is setted to the upper limit allowed by the manufacturer.

Table 5. PI Control Performance and Robustness

M_s^t (β)	J_{IAEr}	J_{IAEr}(%)	TV_{ur}	TV_{ur}(%)	M_s^r
2.0 (0.667)	0.363	100%	0.398	81%	2.009
2.0 (1.0)	0.363	100%	0.489	100%	2.009
1.2 (2.501)	0.471	66%	0.201	101%	1.239
1.2 (1.0)	0.717	100%	0.200	100%	1.239

5. CONCLUSIONS

The use of a Two-Degree-of-Freedom (2-DoF) PID controller must allow the control-loop designer to take into consideration the *regulatory control performance* and *control effort* requirements in conjunction with the *control system robustness* and then improve the *servo-control performance*.

However the analysis of the recommended tuning for its proportional set-point weight has shown that the established constraint by controller's manufactures for its values to the $0 \leq \beta \leq 1.0$ range, avoids the designer to exploit the full potential of these controllers.

The allowed range for the proportional set-point weight could make sense when the controller design main objectives were only to optimize its performance but nowadays, the *performance-robustness trade-off* is taken into account within the *modern* control design formulations. Even included explicitly into the tuning equations as it has been shown in the concrete tuning rules analyzed in this paper. It has been shown that this constraint reduces the performance of the control-loop responses, to a set-point step change, when a high robustness control system is required.

The control system designer will be able to use the full inherent capabilities of the Two-Degree-of-Freedom PID controllers, only when the existing constraint in the set-point weight selection will be removed by the manufacturers.

ACKNOWLEDGEMENTS

This work has received financial support from the Spanish CICYT program under grant DPI2007-63356.

The financial support from the University of Costa Rica and from the MICIT and CONICIT of the Government of the Republic of Costa Rica is greatly appreciated.

REFERENCES

ABB ((n.d.)). *Protonic 100/500/550 Digitric 500 Configuration and Parametrization Manual*. ABB Automation Products. 42/62-500 12EN.

Alfaro, V.M., Vilanova, R., and Arrieta, O. (2008). Analytical Robust Tuning of PI controllers for First-Order-Plus-Dead-Time Processes. In *13th IEEE International Conference on Emerging Technologies and Factory Automation*. Hamburg-Germany.

Araki, M. (1984a). On Two-Degree-of-Freedom PID Control System. Technical report, SICE Research Commitee on Modeling and Control Design of Real Systems.

Araki, M. (1984b). PID Control Systems with Reference Feedforward (PID-FF Control System). In *Proc. of 23rd SICE Anual Conference*, 31–32.

Araki, M. (1985). Two-Degree-of-Freedom Control System - I. *Systems and Control*, 29, 649–656.

Arrieta, O. and Vilanova, R. (2007a). Performance degradation analysis of Optimal PID settings and Servo/Regulation *tradeoff* tuning. *CSC07, Conference on Systems and Control, Marrakech-Morocco*.

Arrieta, O. and Vilanova, R. (2007b). PID Autotuning settings for balanced servo/regulation operation. *MED07, 15th IEEE Mediterranean Conference on Control and Automation, Athens-Greece*.

Arrieta, O. and Vilanova, R. (2007c). Servo/Regulation tradeoff tuning of PID controllers with a robustness consideration. *CDC07, 46th IEEE Conference on Decision and Control, New Orleans, Louisiana-USA*.

Arrieta, O., Vilanova, R., Alfaro, V., and Moreno, R. (2008). Considerations on PID Controller Operation: Application to a Continuous Stirred Tank Reactor. In *13th IEEE International Conference on Emerging Technologies and Factory Automation*. Hamburg-Germany.

Åström, K. and Hägglund, T. (1995). *PID Controllers: Theory, Design and Tuning*. Instrument Society of America, Research Triangle Park, NC, USA.

Åström, K. and Hägglund, T. (2004). Revisiting the Ziegler-Nichols step respose method for PID control. *Journal of Process Control*, 14, 635–650.

Åström, K. and Hägglund, T. (2006). *Advanced PID Control*. ISA - The Instrumentation, Systems, and Automation Society.

Åström, K., Hang, C.C., Persson, P., and Ho, W.K. (1992). Towards Intelligent PID Control. *Automatica*, 28(1), 1–9.

Åström, K., Panagopoulos, H., and Hägglund, T. (1998). Design of PI controllers based on non-convex optimization. *Automatica*, 34, 585–601.

Babb, M. (1990). Pneumatic Instruments Gave Birth to Automatic Control. *Control Engineering*, 37(12), 20–22.

Bennett, S. (2000). The Past of PID Controllers. In *IFAC Digital Control: Past, Present and Future of PID Control*. Terrassa, Spain.

Chien, I., Hrones, J., and Reswick, J. (1952). On the automatic Control of generalized passive systems. *Trans. ASME*, 175–185.

Cohen, G.H. and Coon, G.A. (1953). Theoretical Considerations of Retarded Control. *ASME Transactions*, 75, Jul.

Emerson (2008). *DeltaV BooksOnLine9.3*. Emerson Process Managment. Web-based version http://www.easydeltav.com/BOL/.

Foxboro (1998). *Instruction 762 CNA Single Station Micro Controller*. The Foxboro Company. MI 018-885.

Fuji (2001). *Instruction Manual - Compact Controller M*. Fuji Electric. INP-TN1PDA 3cE.

Gorez, R. (2003). New desing relations for 2-DOF PID-like control systems. *Automatica*, 39, 901–908.

Hägglund, T. and Åström, K. (2002). Revisiting the Ziegler-Nichols tuning rules for PI control. *Asian Journal of Control*, 4, 354–380.

Hang, C. and Cao, L. (1996). Improvement of Transient Response by means of variable set point weighting. *IEEE Transaction on Industrial Electronics*, 4, 477–484.

Honeywell (2007). *UDC 3500 Universal Digital Controller Product Manual*. Honeywell International. 51-52-25-120.

Horowitz, I.M. (1963). *Shynthesis of Feedback Systems*. Academic Press.

Kaya, I. (2004). Tuning PI controllers for stable process with specifications on gain and phase margings. *ISA Transactions*, 43, 297–304.

López, A.M., Miller, J.A., Smith, C.L., and Murrill, P.W. (1967). Tuning Controllers with Error-Integral Criteria. *Instrumentation Technology*, 14, 57–62.

Martin, J., Smith, C.L., and Corripio, A.B. (1975). Controller Tuning from Simple Process Models. *Instrumentation Technology*, 22(12), 39–44.

Méndez, V. (2008). *Performance and Robustness of PID Control Loops*. Licenciatura Thesis, Escuela de Ingeniería Eléctrica, Universidad de Costa Rica. (in Spanish).

Mitsubishi (2002). *System Q Programmable Controller Logic Control Programming Manual*. Mitsibishi Electric. 149256.

O'Dwyer, A. (2003). *Handbook of PI and PID Controller Tuning Rules*. Imperial College Press, London, UK.

Omron (2007). *Instructions Reference Manual Programmable Controllers SYSMAC CS Series*. Omron Electronics LLC.

Rivera, D.E., Morari, M., and Skogestad, S. (1986). Internal Model Control. 4. PID Controller Desing. *Ind. Eng. Chem. Des. Dev.*, 25, 252–265.

Rockwell (2003). *MicroLogix 1200 and 1500 Programmable Controllers Bulletin 1762 and 1764 Instruction Set Reference Manual*. Rockwell Automation.

Rockwell (2005). *Logix 5000 Controllers Process Control and Drivers Instructions User Manual*. Rockwell Automation. PN 957955.

Rovira, A., Murrill, P.W., and Smith, C.L. (1969). Tuning Controllers for Setpoint Changes. *Instrumentation & Control Systems*, 42, 67–69.

Sung, S.W. and Lee, I.B. (1999). *PID Controllers and Automatic Tuning*. A-JIN Publishing Co., Seul, Korea.

Taguchi, H. and Araki, M. (2000). Two-Degree-of-Freedom PID controllers - Their functions and optimal tuning. In *IFAC Digital Control: Past, Present and Future of PID Control*. Terrassa, Spain.

Taguchi, H. and Araki, M. (2002). Survey of researches on Two-Degree-of-Freedom PID controllers. In *The 4th Asian Control Conference*. Singapore.

Taguchi, H., Kokawa, M., and Araki, M. (2002). Optimal tuning of two-degree-of-freedom PD controllers. In *The 4th Asian Control Conference*. Singapore.

Visioli, A. (2006). *Practical PID Control*. Springer Verlag Advances in Industrial Control Series.

Yokogawa ((n.d.)). *YS 1500 Indicating Controller User Manual*. Yokogawa Electric Company. IM 01B08B01-02E.

Ziegler, J.G. and Nichols, N.B. (1942). Optimum Settings for Automatic Controllers. *ASME Transactions*, 64, 759–768.

A nonlinear control strategy for a Bidirectional Flow Process

Pablo Zúñiga Salas Héctor Ramírez Estay Daniel Sbarbaro Hofer

Department of Electrical Engineering, Universidad de Concepción,
Concepción, Chile (Tel: +56 41 2204353; e-mail: pablozuniga, hectrami, dsbarbar @udec.cl).

Abstract: A nonlinear control strategy based on Interconnection Damping Assignment Passivity Based Control (IDA-PBC) is proposed for a process with bidirectional flow. The bidirectional flow condition introduces singularities in the control action under certain operation conditions. A solution to this problem is proposed such that operation through the singular points is possible and the stability conditions around the desired operation point are exactly preserved. In addition, a passivity based integral action is included in order to take into account the effects of model uncertainties and unknown step like disturbances. A description of the process and the controller design methodology is presented along with some numerical simulations illustrating the closed-loop behavior of the proposed controller.

1. INTRODUCTION

There are many applications where the characteristic of the process changes and there is the possibility of having singular points associated to the control variable. This condition means that the states of the process are not controllable at the singular point and the control variable became unbounded. This condition can be found for instance in chemical reactors (E.J. McColm and M. T. Tham, 1995) and in electromechanical systems (F. Zhang and B. Fernandez, 2006). In most cases this problem is overcome by modifying the control law such that the singular point is eliminated. There are two approaches for dealing with this problem: the first one is based on differentiation (E. J. McColm and Ming T. Tham, 1995), and the second one on a modification of the control law (H. Xu and P.A. Ioannou, 2004). In this work, the second option is used to design an IDA-PB controller that can deal with singular points without affecting the closed loop stability when the system is outside the set of singular points.

Port-Hamiltonian (PH) systems and Interconnection Damping Assignment Passivity Based Control (IDA-PBC) are two powerful approaches for modeling and control of nonlinear systems. PH representations are physical motivated, since they are based on models representing mass and energy balances; where the structure of the model takes into account the interaction between the system and it environment.

IDA-PBC control approach relies on the notions of interconnections, dissipation and energy balance, (Ortega et al., 2001, 2002). The capability to define precisely the interconnections and energy dissipations of non-linear processes makes the use of PH representation an attractive modeling tool and hence, IDA-PBC an effective alternative to design high performance non-linear controllers for complex processes.

The IDA-PBC approach has been shown to be very effective when the system characteristics changes from one operation mode to another. For instance, in (Ramírez et al., 2008) the same IDA-PB controller is used to stabilize a minimum and a non-minimum phase system, and in (Batlle et al., 2005) this approach is used to stabilize a bidirectional power systems, where the direction of power is reversed depending on the operation mode of a power converter. In this work, a nonlinear bidirectional flow process with singular points operation is used as an application example. The process consists of three serials tanks at same height; hence flow inversion between tanks is possible. The process also considers an unknown disturbance.

This paper is organized as follows: Section 2 describes a process comprising three tanks in series with the possibility of having reversing flow, and the model of this system. Section 3 presents the design of a IDA-PBC plus integral action. In section 4, a solution to the singular point operation is proposed and the system stability is analyzed. Some simulations results are presented in section 5 and finally, in section 6, some closing remarks are given.

2. THE THREE TANKS CONTROL PROBLEM

The multi tank serial circuit is a multivariable system, fully actuated and minimum phase. This process has three tanks of the same height in a serial arrangement, as depicted in Fig. 1.

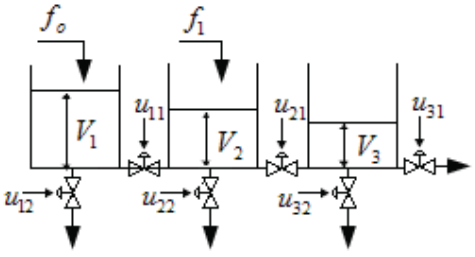


Fig. 1. Proposed system of serial tanks

In Fig. 1 the control valves are u_{11}, u_{21} and u_{31}. The remaining valves, u_{12}, u_{22} and u_{32} are manual valves, whose openings remain constant during the entire operation. On other hand, the feed flow rate into the first tank is measurable and the feed flow rate into the second tank is unknown. The last tank does not have any independent feed flow rate.

The control objective is to operate the tanks at different heights by allowing flow reversing operations.

The tanks are at the same height so the flow direction between tanks depends on the bottom pressure on each of them; i.e. the flow will go from the tank with higher water level to the tank with lower one. This reversing flow phenomenon occurs only in the first and second tanks because they are the only ones with an independent feed flow rate, f_o and f_1 respectively. The maximum water level in the third tank is the level of the second tank, because it does not have an independent water flow rate. Another phenomenon, that arises during the flow reversing process, is the lack of controllability, this occurs when the water level in the first and second tank are the same.

Physically, given a set of feed flow rates the processes always operate in one of these modes. In order to model the process using mass balance equations, the following variables are defined: $x_i \in \mathbb{R}_+$ is the volume inside of a tank *i*, A_i the cross section (they are constant and the same for all tanks) and $k_{ij}(u_{ij}) = u_{ij}$ linear valve opening functions. Thus, the equations representing the system are:

$$\dot{x}_1 = f_o - u_{11} \cdot sign(x_1 - x_2) \cdot a_1 - b_1 \,, \tag{1}$$

$$\dot{x}_2 = f_1 + u_{11} \cdot sign(x_1 - x_2) \cdot a_1 - u_{21} \cdot sign(x_2 - x_3) \cdot a_2 - b_2 \,, \tag{2}$$

$$\dot{x}_3 = u_{21} \cdot sign(x_2 - x_3) \cdot a_2 - u_{31} \cdot a_3 - b_3 \,, \tag{3}$$

where, for notational convenience, we have defined:

$$a_1 = \sqrt{2g\left(\frac{x_1}{A_1} - \frac{x_2}{A_2}\right)} \;;\; a_2 = \sqrt{2g\left(\frac{x_2}{A_2} - \frac{x_3}{A_3}\right)} \;;\; a_3 = \sqrt{2g\frac{x_3}{A_3}}$$

$$b_1 = \sqrt{2g\frac{x_1}{A_1}} \cdot u_{12} \;;\; b_2 = \sqrt{2g\frac{x_2}{A_2}} \cdot u_{22} \;;\; b_3 = \sqrt{2g\frac{x_3}{A_3}} \cdot u_{32} \,.$$

In order to invert the flow rate direction, between the first and second tank, the feed flow rate in the first tank must satisfy $f_o \le b_1$. This can be obtained by calculating the operation point for u_{11}

$$u_{o11} = \frac{f_o - b_1}{a_1 \cdot sign(x_{o1} - x_{o2})} \,. \tag{4}$$

Thus, in order to have $u_{o11} \ge 0$, for a operation point $x_{o1} < x_{o2}$ with $x_{o1}, x_{o2} \in \mathbb{R}_+$, the following inequality has to be satisfied that $f_o - b_1 \le 0$. To get a flow from the first to the second tank we need $f_o - b_1 \ge 0$.

We will also define the set of singular points as all the $x_1, x_2 \in \mathbb{R}_+$, such that $x_1 = x_2$.

From the knowledge of the process and given a set of possible combinations of feed flow rates, the following operating modes can be identified:

Operation Mode 1: The feed flows are: $f_o > b_1, f_1 \ge 0$, the initial state $x_1^0 > x_2^0 > x_3^0$, with $x_1^0, x_2^0, x_3^0 \in \prod \subset \mathbb{R}_+$ and references $x_1^* > x_2^* > x_3^*$, with $x_1^*, x_2^*, x_3^* \in \mathbb{R}_+$. The set $\prod$ represents the physical admissible levels. The flow goes from the first to the second tank.

Operation Mode 2: The feed flows are: $f_o \le b_1, f_1 > 0$, the initial state $x_1^0 > x_2^0 > x_3^0$, with $x_1^0, x_2^0, x_3^0 \in \prod \subset \mathbb{R}_+$ and references $x_1^* < x_2^*$ and $x_3^* < x_2^*$, with $x_1^*, x_2^*, x_3^* \in \mathbb{R}_+$. The flow is inverted and goes from the second to the first tank.

Operation Mode 3: The feed flows are: $f_o \le b_1, f_1 > 0$, the initial state $x_1^0 < x_2^0$ and $x_3^0 < x_2^0$, with $x_1^0, x_2^0, x_3^0 \in \prod \subset \mathbb{R}_+$ and references $x_1^* < x_2^*$ and $x_3^* < x_2^*$, with $x_1^*, x_2^*, x_3^* \in \mathbb{R}_+$. The flow goes from the second to the first tank.

Operation Mode 4: The feed flows are: $f_o > b_1, f_1 \ge 0$, the initial state $x_1^0 < x_2^0$ and $x_3^0 < x_2^0$, with $x_1^0, x_2^0, x_3^0 \in \prod \subset \mathbb{R}_+$ and references $x_1^* > x_2^* > x_3^*$, with $x_1^*, x_2^*, x_3^* \in \mathbb{R}_+$. The flow is inverted and goes from the first to the second tank.

3. CONTROLLER DESIGN USING IDA-PBC

It is convenient to represent the system in a PH form, to simplify the application of the IDA-PBC.

Consider a process described by a PH system of the form

$$\dot{x} = \left[J(x) - \Re(x)\right]\frac{\partial H}{\partial x}(x) + g(x)u + q(x)f \tag{5}$$

Where $x \in \mathbb{R}^n$ and $u \in \mathbb{R}^m$ are the mass (volume) variables and the control, respectively. The smooth function $H(x)$ typically represents the total stored mass, $f = \begin{bmatrix} f_o & f_1 \end{bmatrix}^T \in \mathbb{R}^m$ represents constant disturbances and $q(x)$ defines the interaction between the system and *f*. the skew-symmetric matrix $J(x) = -J^T(x)$ represents the interconnection between the different system's components, and $\Re(x) = \Re^T(x) \ge 0$ is the dissipation matrix, while $g(x)$ defines the interconnection of the system with its

environment. A detailed overview of PH systems can be found in (van der Shaft, 2004).

To represent the tank processes as PH model, the following storage function is selected, which represents the total mass (volume) in the system

$$H(x) = x_1 + x_2 + x_3 \geq 0\,. \tag{6}$$

The IDA-PBC methodology allows to find a static control feedback $u = \beta(x)$ such that the desired performance is specified by a closed loop dynamic as

$$\dot{x} = \left[J_d(x) - \Re_d(x)\right]\frac{\partial H_d(x)}{\partial x}, \tag{7}$$

where $H_d(x)$ is the desired total mass function fixed by the designer and which has a strict minimum in x^*. The matrices $J_d(x) = -J_d{}^T(x)$ and $\Re_d(x) = \Re_d{}^T(x) \geq 0$ are the desired interconnection and damping matrices respectively. In order to get decoupled outputs, the closed loop port Hamiltonian system has to have a null interconnection matrix and a diagonal damping matrix. For accomplishing this objective, it is possible to define the open-loop PH system such that it satisfies these characteristics. The PH matrixes are

$$J(x) = -J^T(x) = 0\,, \tag{8}$$

$$\Re(x) = \Re^T(x) = diag\{b_1, b_2, b_3\}\,, \tag{9}$$

$$g(x) = \begin{bmatrix} -sign(x_1 - x_2)\cdot a_1 & 0 & 0 \\ sign(x_1 - x_2)\cdot a_1 & -sign(x_2 - x_3)\cdot a_2 & 0 \\ 0 & sign(x_2 - x_3)\cdot a_2 & -a_3 \end{bmatrix}, \tag{10}$$

$$q(x) = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \end{bmatrix}^T. \tag{11}$$

The closed loop interconnection and damping matrices have to be equal to the open loop matrices; i.e. $J_d(x) = J(x)$ and $\Re_d(x) = \Re(x)$. If this is satisfied, the closed-loop process has decoupled outputs. The references levels are constant, so the following desired storage function (Ramírez *et al,* 2008) can be used,

$$H_d(x) = \sum_{i=0}^{n} x_i + \sum_{i=0}^{n} -(1-k_i)x_i - k_i x_i^* \ln(x_i)$$

$$= \sum_{i=0}^{n}\left[x_i - k_i x_i^* \ln(x_i)\right]. \tag{12}$$

For the given desired storage function and for previously defined open and closed loop PH matrices, by matching (5) and (7) the following control law, where $\hat{f}_1$ is an estimation of f_1, is obtained

$$u_{11} = \frac{f_o - b_1 + b_1 \cdot k_1\left(1 - \frac{x_1^*}{x_1}\right)}{sign(x_1 - x_2)\cdot a_1}, \tag{13}$$

$$u_{21} = \frac{\hat{f}_1 - b_2 + b_2 k_2\left(1 - \frac{x_2^*}{x_2}\right) + u_{11}\cdot a_1 \cdot sign(x_1 - x_2)}{sign(x_2 - x_3)\cdot a_2}, \tag{14}$$

$$u_{31} = \frac{-b_3 + b_3 k_3\left(1 - \frac{x_3^*}{x_3}\right) + u_{21}\cdot a_2 \cdot sign(x_2 - x_3)}{a_3}\,, \tag{15}$$

If the control inputs (13), (14) and (15) are replaced in (5), then the time derivate of the desired storage function, will be negative, thereby the closed-loop system is asymptotically stable.

Equations (13) and (14) require to know the system parameters and flow rates f_o and f_1. In order to compensate the lack of knowledge about the values of these flow rates, an integral action is considered in the final control law. The solution used in this paper was presented in (Ortega and García-Canseco, 2004) and consists in adding an integral term of the passive output to the control.

Let's consider the system presented in (5) in closed loop with $u = \beta(x) + \upsilon$, where υ is an integral action added to the system trough a state variable and defined as

$$\dot{\upsilon} = -K_I g^T(x)\nabla H_d(x) \tag{16}$$

With $K_I = K_I{}^T > 0$. Then, all stability properties of x^* are preserved. In fact the closed loop clearly takes the PCH form

$$\begin{bmatrix} \dot{x} \\ \dot{\upsilon} \end{bmatrix} = \begin{bmatrix} J_d(x) - \Re_d(x) & g(x)K_I \\ -K_I g^T(x) & 0 \end{bmatrix}\begin{bmatrix} \nabla_x W \\ \nabla_\upsilon W \end{bmatrix}, \tag{33}$$

where

$$W(x,\upsilon) = H_d(x) + \frac{1}{2}\upsilon^T K_I{}^{-1}\upsilon \tag{17}$$

is the new total storage function which now qualifies as Lyapunov function. For this application it is convenient to use a diagonal gain matrix, i.e. $K_I = diag\ \{k_{I1}, k_{I2}, k_{I3}\}$, $g(x)$ like in (10) and $H_d(x)$ like in (12).

4. SINGULAR POINT REGULARIZATION

The control action (13), (14) and (15), have singulars points arising when two tanks have the same level. In this case, the flow between these tanks becomes null and the control action becomes inexistent. Operating the system on this singulars point is not required in this application. However, if the controller attempts to invert the flow between two contiguous tanks, it is necessary to pass from a state $x_1 > x_2$ to a state

$x_1 < x_2$. Along the trajectory is necessary to pass trough $x_1 = x_2$, which make the control law unfeasible. This only happens between the first and second tank, hence, the solution is only used in the first control input.

Based on the work of Haojian and Ioannou (2004), a singular point solution is proposed. Assume a function $\eta(x) \in \mathbb{R}$ such that $\eta(0) = 0$. The inverse of $\eta(x)$ is undetermined at zero. To avoid this, the following solution is proposed:

$$\frac{1}{\eta(x)} \Rightarrow \frac{\eta(x)}{\eta^2(x) + \delta(x)}, \tag{18}$$

where $\delta(x)$ is defined as follows:

$$\delta(x) = c_1 \cdot \left(1 - \frac{1}{1 + e^{-c_2 \cdot |x_1 - x_2|}}\right) \cdot \left| e_{ctrl}(x, x^*) \right| \cdot k_{sp}, \tag{19}$$

where k_{sp}, c_1 and c_2 are real positive constants and they are considered as tuning parameters. Equations (19) means that $\delta(x)$ will be zero if x is at the reference x^*. Of course, the references x_1^* and x_2^* must be different, otherwise (18) will be unbounded.

The control input (13) including the singular point solution becomes:

$$u_{11} = \frac{sign(x_1 - x_2) \cdot a_1 \cdot \left(f_o - b_1 + b_1 \cdot k_1 \left(1 - \frac{x_1^*}{x_1}\right)\right)}{a_1^2 + \delta(x)}. \tag{20}$$

The control variables u_{21} and u_{31} do not require changes since they do not have singular points.

4.1 Stability Analysis

In this section a brief and simple stability analysis is carried out. Replacing (20), (14) and (15) in (1), (2) and (3), respectively, the closed loop system takes the form:

$$\dot{x}_1 = -\frac{a_1^2}{a_1^2 + \delta(x)} b_1 k_1 \left(1 - \frac{x_1^*}{x_1}\right) + \left(f_o - b_1\right) \frac{\delta_1(x)}{a_1^2 + \delta(x)} \tag{21}$$

$$\dot{x}_2 = -b_2 k_2 \left(1 - \frac{x_2^*}{x_2}\right) + \Delta f_1 \tag{22}$$

$$\dot{x}_3 = -b_3 k_3 \left(1 - \frac{x_3^*}{x_3}\right) \tag{23}$$

where $\Delta f_1 = f_1 - \hat{f}_1$ is the estimation error of the unknown disturbance in the second tank.

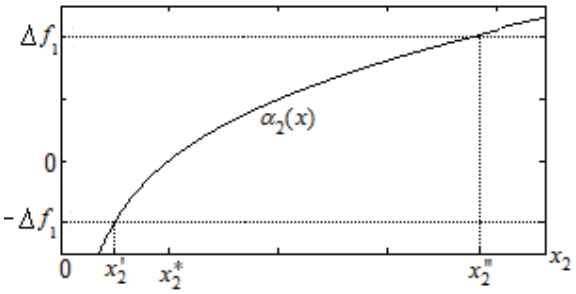


Fig. 2. Right hand terms of equation (22)

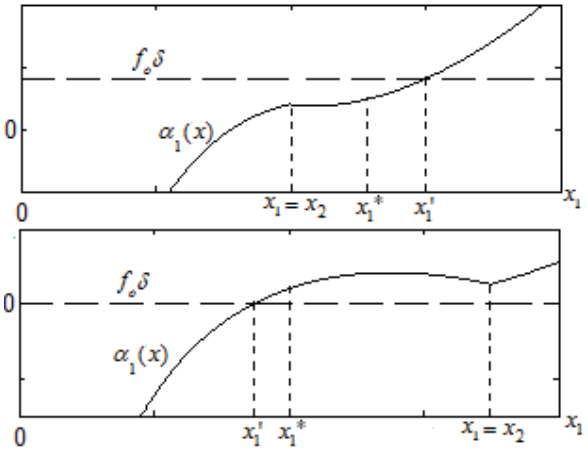


Fig. 3. Right hand terms of equation (21) for a constant δ

Then stability of the closed-loop system can be analyzed as follows: Since equation (23) only depends on x_3 and the term $b_3 k_3$ is positive, x_3^* will be an asymptotically stable equilibrium point. From equation (22) we have that the dynamic only depends on x_2. The stability analysis can be carried out by analyzing the right hand terms, as they are depicted in Fig. 2., where Δf_1 and the term $\alpha_2(x) = b_2 k_2 (1 - x_2^*/x_2)$ have been drawn in terms of x_2. From this plot can be seen that the system will converge to a unique equilibrium point x_2'', so that $\|x_2'' - x_2^*\| < \gamma$, where $\gamma(\Delta f_1) > 0$ is a real positive constant that depends on Δf_1. If $\Delta f_1 = 0$, then x_2^* will be asymptotically stable equilibrium point, as x_3^*. The analysis for x_1 consider the perturbation term $(f_o - b_1)\delta/(a_1^2 + \delta(x))$, which vanishes at the equilibrium, hence x_1 could converge asymptotically to x_1^*. In fact, the expression for the equilibrium point is:

$$\left(f_o - b_1\right)\delta(x) - a_1^2 b_1 k_1 \left(1 - \frac{x_1^*}{x_1}\right) = 0. \tag{24}$$

Equation (24) can be verified at the desired operation point x_1^* or if $(f_o - b_1) = 0$ and $a_1 = 0$. The last case can only be possible if the system is at the singular point; i.e. if $x_1^* = x_2^*$ and $f_o = b_1 = (2g x_1)^{1/2}$. Fig. 3 depicts the right hand terms of (21); $\alpha_1(x)$ defined as:

$$\alpha_1(x) = b_1 \delta(x) + a_1^2 b_1 k_1 \left(1 - \frac{x_1^*}{x_1}\right), \tag{25}$$

and for both condition $(f_o - b_1) > 0$ and $(f_o - b_1) < 0$. From this figure can be seen that there exist only one asymptotically stable equilibrium point x_1^*, which is not the desired reference value, the steady state error will depend on δ and f_o. Fig. 5, shows the effect of making δ dependant on the variable x, as in (20). In this case, the desired reference value is an asymptotically stable equilibrium point.

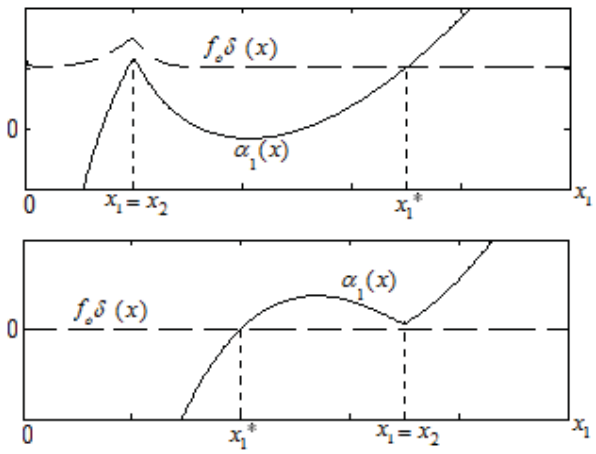


Fig. 4. Right hand terms of equation (21) for a variable δ (x)

5. NUMERICAL SIMULATIONS

In this section, some simulation results illustrating the controller characteristics are presented. The tuning parameters were selected to obtain a closed loop response with overshoots smaller than 15% and small settling times according with the open loop dynamics.

This simulation considers the following: linear control valves, i.e. $k_{ij}(x) = u_{ij}$ and the cross section of all tanks are the same and constant, i.e. $A_1 = A_2 = A_3 = 1731.3$. The tuning parameters for the singular points solution where selected as $C_1 = 500$ y $C_2 = 0.0004$. The feed flows rates have their maximum value at 4000 cm^3/s and are represented in percentage values.

If the flow direction, between the first and second tank, is inverted, then the system go through a singular point. The following simulations show the performance of the system with a flow rate inversion. The parameters were $k_1=4$, $k_2=2$, $k_3=1$ and the integrator parameter were set at $k_{I1}=0.0001$, $k_{I2}=0.01$, $k_{I3}=0.01$.

In the first part, the system is working with a level of 10cm in the first tank and 5 cm in the second one, and the flow between the tanks goes from the first to the second. Fig.5 depicts the closed loop behavior. At 4250 the both set points were increased at the same time. to 35cm and 25cm respectively. At 5000 sec. the level reference of the first tank is set to 15cm and for the second one is kept at 25cm, leading to an inversion of the flow direction. From Fig. 5 can be seen that while the level in the first tank changes, the control tries to maintain the level in the second tank constant, and it invert the flow direction without discontinuities in the control inputs. The coupling between the outputs is due to the integrator compensation, since the static feedback was designed considering a null interconnection matrix. These good results are achieved due to the joint action of the integral action and the methodology used to deal with the singular point.

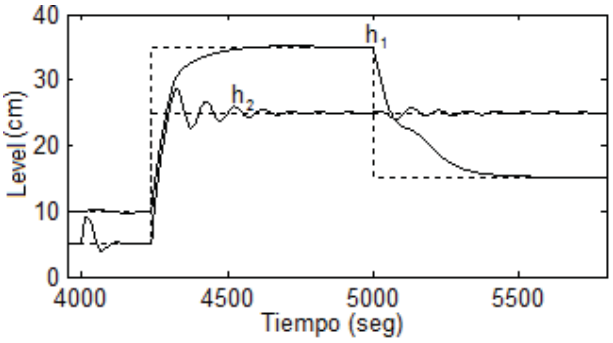


Fig. 5. Levels in first and second tank. Integral action and flow rate inversion

From Fig. 6, can be seen that the control input, in both tanks (1 and 2), are smooth, continuous and bounded. Beside, Fig. 7 shows the flow rate inversion (5070 seconds approx.), and the sudden changes of the flow rate when the control inputs changes their values due to references changes.

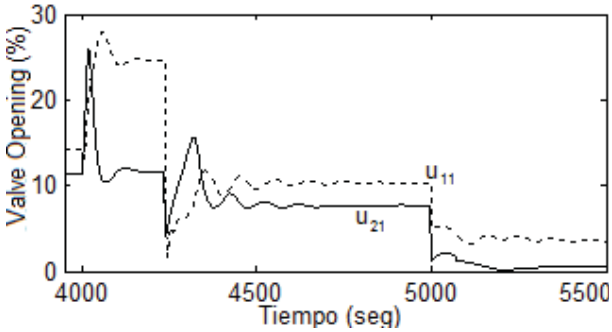


Fig. 6. Control input of the first and second tank. Integral action and flow rate inversion

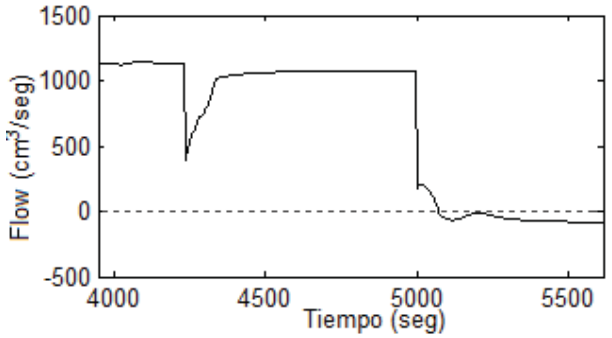


Fig. 7. Flow rate between the first and second tank. Integral action and flow rate inversion

A simple PI controller can not deal with reversing flows, since the process open-loop gain changes sign when the flow changes direction.

6. CONCLUDING REMARKS

This paper presents a novel nonlinear control strategy based on IDA-PBC for a non-linear process with bidirectional flow. The process was modeled as PH model, and by a proper selection of the process closed-loop matrices. A passivity based strategy was designed, and in order to deal with model uncertainties and unknown disturbances, integral action was also considered in the control law. Since the process exhibits uncontrollable operation conditions; i.e. singular points in the control law, a singular point solution was proposed without compromising the stability conditions of the closed-loop process. A nice feature of the proposed singular point solution is that outside the set of singular points the desired closed-loop interconnection and damping specifications are preserved, hence no special considerations must be taken into account when selecting the desired closed-loop PH system in the IDA-PBC design. The closed-loop behavior of the proposed controller has been illustrated by numerical simulations.

Future works will consider a more detailed stability analysis for the general case, including integral actions. Implementation of the controller in a laboratory application is also part of the future work to be carried out.

ACKNOWLEDGEMENTS

This work was supported by Fondecyt Project 1070491

REFERENCES

E. J. McColm and M. T. Tham (1995). "On Globally Linearising Control about Singular Points", *Proceedings of the American control conference*, pp. 2229-2233.

R. Ortega, A. van der Schaftb, I. Mareels and B. Maschke. (2001). "Putting Energy Back in Control", *Control Systems Magazine, IEEE*, vol. 21, pp. 18-33.

R. Ortega, A. van der Schaft, B. Maschke, G. Escobar (2002). "Interconnection and damping assignment passivity-based control of port controlled Hamiltonian systems". Automatica 38, pp. 585-596.

R. Ortega and E. García-Canseco (2004). "Interconnection and damping assignment passivity based control: A survey", *European Journal of Control*, vol. 10, pp. 432-450.

H. Xu and P. Ioannou (2004). "Robust Adaptive Control of Linearizable Non-linear Single Input Systems with Guaranteed Error Bounds", *Automatica,* vol. 40, pp. 1905-1911.

A.J. van der Schaft (2004). "Port-Hamiltonian systems: network modeling and control of nonlinear physical systems", *In: H. Irschik, K. Schlacher (Eds.), Advanced Dynamics and Control of structures and Machines, pp. 127–168.*

C. Batlle, A. Doria - Cerezo and E. Fossas (2005). "IDA-PBC Controller for a Bidirectional Power Flow Full-Bridge Rectifier", *Decision and Control, 2005 and 2005 European Control Conference. CDC-ECC '05. 44th IEEE Conference on*, pp. 422-426.

J. Johnsen and F. Allgöwer (2006). "Interconnection and Damping Assignment Passivity-Based Control of a Four-Tank System", *F. Bullo y K. Fujimoto (Eds.), Preprints of the IFAC 3rd Workshop on Lagrangian and Hamiltonian Methods for Nonlinear Control*, vol. 60, pp. 69–74.

F. Zhang and B. Fernandez, 2006. "Feedback Linearization control of System with Singularities: A Ball Beam Revisit", *International conference on Complex Systems (ICCS2006).*

H. Ramírez, D. Sbarbaro and R. Ortega (2008). "On the control of non-linear processes: An IDA–PBC approach", *Journal of Process Control*, vol 19, pp. 405-414.

J.K. Johnsen, F. Döorer, and F. Allgöwer (2008). "L2-gain of Port-Hamiltonian systems and application to a biochemical fermenter". *Proceedings of the American Control Conference 2008*, pages 153-158.

Characteristics-based MPC of a fixed bed reactor with catalyst deactivation

Leily Mohammadi * **Ilyasse Aksikas** * **J. Fraser Forbes** *

* *Chemical and Materials Engineering department, University of Alberta, Edmonton, AB Canada*
(e-mail: {leily, aksikas, fraser.forbes }@ualberta.ca).

Abstract: In this work characteristics-based model predictive control (CBMPC) of a fixed bed reactor with catalyst deactivation is studied. Performance of CBMPC has been analyzed for two cases: one that incorporates the catalyst deactivation within the reactor model and another that ignores the deactivation. Simulation results show that the performance of first controller that incorporates the catalyst deactivation is better than the controller that ignores the deactivation.

Keywords: Fixed bed reactor, Catalyst deactivation, Characteristics-based model predictive control

1. INTRODUCTION

A catalyst loses its activity during operation. Catalyst deactivation can have variety of consequences. It may cause thermal instability of the reactor. It also affects the conversion and selectivity of the desired reaction. Consequently, it will affect the productivity and energy efficiency of the plant. In order to compensate for the effect of catalyst deactivation, the operating conditions are changed gradually to ensure maintaining the quality of product and the rate of production. Then designing a controller that can ensure changing optimal process operating conditions are tracked as they vary, is an important issue in operation of catalytic reactors.

Integration of the catalyst deactivation dynamics into the reaction system model results in a model that can describe the dynamical behavior of the system more precisely. By using this model in model-based algorithms, one can design a more efficient controller.

The objective of this work is to study the control of a fixed bed reactor with catalyst deactivation. In order to capture all of the main "macroscopic" phenomena (i.e., reactions, diffusion, convection, and so forth), the model of a fixed bed reactor takes the form of a mixed set of partial differential, ordinary differential, and algebraic equations. Such systems and many others (e.g., systems modeled by partial difference equations, integral equations and delay differential equations) are called distributed parameter systems (DPS) or infinite dimensional systems. Since we will consider the catalyst deactivation in the reactor's model, the resulting infinite dimensional system will be time varying.

Aksikas et al. (2009) and Aksikas et al. (2008) studied the control of the time varying infinite dimensional systems. In these works linear-quadratic controllers are developed by solution of the classical Riccati equation. This work is extended by Mohammadi et al. (2009) to cover the two-time scale property of the fixed bed reactors.

Model Predictive Control (MPC) is an optimal control technique that uses a model of the system to predict the future plant behavior and determines a sequence of control moves so that the predicted response moves to the desired set point in an optimal manner. Unfortunately, MPC algorithms for distributed parameter systems are relatively scarce. For diffusion-reaction systems, which are described by parabolic PDEs, Dubljevic et al. (2005) used modal decomposition to derive finite-dimensional systems that capture the dominant dynamics of the original PDE and are subsequently used for controller design. For the convection dominated parabolic PDEs, the modal decomposition methods result in high-order finite dimensional systems. MPC for high-order systems is computational demanding and cannot be applied on-line. For hyperbolic systems, the eigenvalues of the spatial differential operator cluster along vertical or nearly vertical asymptotes in the complex plane[Christofides (2001)], and the modal decomposition methods may not be used. Dubljevic et al. (2005) used the finite difference method to convert the hyperbolic equations to a set of ODEs and the MPC is designed for the resulting model. Using discretization methods may result in improperly capturing the dynamics of the system. More-

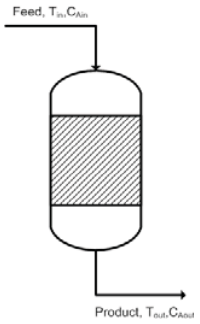


Fig. 1. Schematic diagram of Fixed-Bed reactor

over, the resulting ODEs are high-order and may result in an MPC that has high computational computational requirements.

Characteristics-based MPC is an approach for model predictive control of DPS proposed by Shang et al. (2004) and Shang et al. (2007). The method of characteristic allows controller design for linear, quasilinear, nonlinear low dimensional PDEs. In this method, partial differential equations are transformed to a set of ordinary differential equations along the characteristic curves, which exactly describe the original DPS. Then the controller design can be performed on ODEs instead of PDEs without approximation.

The process considered in this work is a catalytic hydrotreating reactor. Hydrotreating is the conventional means for removing sulfur from petroleum fractions. A schematic diagram of this reactor is shown in Fig.1. An important feature of a fixed bed reactor is the two time scale property of the system. In the other words, the dynamic behavior of the material balance is faster than the energy dynamics. Due to this property, the system has two characteristic curves. Furthermore, by incorporating the catalyst deactivation equation within reactor's model another very slow dynamic will be added to the system.

In this work the problem of controlling a fixed-bed catalytic reactor with catalyst deactivation is considered. We applied nonlinear characteristic-based MPC on-line algorithm to control the temperature of the reactor at the desired setpoint during the reactor's operation. Two cases have been considered. In the first one, the designed MPC uses a model of the system that considers the catalyst deactivation. In the second one, the catalyst deactivation is ignored for model predictive control development. Then performance of the two cases has been compared.

2. MODEL DESCRIPTION

The dynamics of a fixed-bed reactor can be described by partial differential equations derived from mass and energy balances. To model the reactor, a plug-flow pseudo-homogeneous model is considered. Moreover, we consider a one-spatial dimension model where there are no gradients in the radial direction. In the simplified system considered here, a lumped reaction kinetics equation was assumed and has the following form (see Chen et al. (2001)):

$$r_A = k(t)e^{(-\frac{E}{RT})}C_A^{n_1}C_H^{n_2} \tag{1}$$

Under the above mentioned assumptions, the dynamics of the process are described by the following energy and mass balance partial differential equations (PDE's).

$$\epsilon\frac{\partial C_A}{\partial t} = -u\frac{\partial C_A}{\partial z} - \rho_B k(t)e^{-\frac{E}{RT}}C_A^{n_1}C_H^{n_2} \tag{2}$$

$$\frac{\partial T}{\partial t} = -u\frac{\partial T}{\partial z} + \frac{\rho_B \Delta H_r}{\rho C_p}k(t)e^{-\frac{E}{RT}}C_A^{n_1}C_H^{n_2} \tag{3}$$

Initial and boundary conditions are:

$$\begin{array}{llll} C_A(0,t) & = C_{A,in}, & C_A(z,0) & = C_{A0}(z), \\ T(0,t) & = T_{in}, & T(z,0) & = T_0(z) \end{array} \tag{4}$$

In the equations above, $C_A, T, \epsilon, \rho_B, \rho, C_p, E, C_H, \Delta H_r, u$ denote the reactant concentration, the temperature, the porosity of the reactor packing, the catalyst density, the fluid density, the heat capacity, the activation energy,the hydrogen concentration, the enthalpy of reaction, and the superficial velocity respectively. k is the pre-exponential factor. Catalysts lose their activity with time and as a result this coefficient varies with time. The parameter k is proportional to the catalyst activity, which is a function of time and the operating conditions and can be described by an ODE (see Furimsky and Massoth (1999)). Here, we assume that the operating conditions are maintained in narrow ranges and in this case k is only a function of time, which can be described by:

$$k = k_0 + k_1 e^{-\alpha t} \tag{5}$$

The above expression for the kinetics of naphtha hydrotreating reaction is in agreement with the observations that after a rapid initial deactivation of the hydrotreating catalyst there is a slow deactivation phase and finally a stabilization of the catalyst activity phase.

3. CHARACTERISTICS-BASED MPC

The method of characteristics is a technique for solving hyperbolic partial differential equations. The idea is that every hyperbolic PDE has characteristic curves along which the dynamics evolve and as a result, the PDE can be represented as an equivalent ODE.

Consider a quasilinear system of first-order equations with two dependent variables ν_1,ν_2 and two independent variables t and x.

$$\begin{aligned} \frac{\partial \nu_1}{\partial t} + a_1\frac{\partial \nu_1}{\partial z} &= f_1(\nu_1, \nu_2, u) \\ \frac{\partial \nu_2}{\partial t} + a_2\frac{\partial \nu_2}{\partial z} &= f_2(\nu_1, \nu_2, u) \end{aligned} \tag{6}$$

if $a_1 \neq a_2$, the system has two different characteristics determined by:

$$\begin{aligned} \text{Characteristic C}_1: \quad & \frac{dz}{dt} = a_1 \\ \text{Characteristic C}_2: \quad & \frac{dz}{dt} = a_2 \end{aligned} \tag{7}$$

along these characteristics dynamic of the system is described by:

$$\begin{aligned} \frac{d\nu_1}{dt} &= f_1(\nu_1, \nu_2, u) \quad \text{along characteristic C}_1 \\ \frac{d\nu_2}{dt} &= f_2(\nu_1, \nu_2, u) \quad \text{along characteristic C}_2 \end{aligned} \tag{8}$$

Then, by using the method of characteristics, the set of partial differential equations (6) is transformed to a set of ODEs along the characteristic curves. This set of ODEs can be used to predict the future behavior of the system.

For a fixed-bed reactor which is modeled by equations (2) and (3) the characteristic curves are:

$$\text{C}_1 = \frac{dz}{dt} = \frac{u}{\epsilon} \tag{9}$$

$$\text{C}_2 = \frac{dz}{dt} = u \tag{10}$$

and the state variables C_A and T are described by the following ODEs along the characteristic curves:

$$\frac{\partial C_A}{\partial t} = -\frac{\rho_B}{\epsilon} k(t) e^{-\frac{E}{RT}} C_A^{n_1} C_H^{n_2} \tag{11}$$

$$\frac{\partial T}{\partial t} = \frac{\rho_B \Delta H_r}{\rho C_p} k(t) e^{-\frac{E}{RT}} C_A^{n_1} C_H^{n_2} \tag{12}$$

The characteristic ODEs are coupled with respect to the two characteristic curves, and the future state variables at one spatial point should be determined by simultaneous integration of both characteristic ODEs along two nonparallel characteristic curves. Fig. 2 illustrates the calculation of the future output variables using method of characteristics. This method for prediction of the future behavior is proposed by Shang et al. (2004). The idea is that at $t = t_k$ the measurements of the state variables are available at discretization points and these measurements are used to determine the value of the state variables at intersections of the characteristic curves. This algorithm provides us with the future values of the output variable. For example for point P we have:

$$C_A(P) = \int_{t(Q)}^{t(P)} f_1(Q) \tag{13}$$

$$T(P) = \int_{t(R)}^{t(P)} f_2(R) \tag{14}$$

where:

$$t(P) = \frac{a_1 t(Q) - 2a_2 t(Q) + a_2 t(R) + Z(R) - Z(Q)}{a_1 - a_2} \tag{15}$$

and a_1 and a_2 are $\frac{u}{\epsilon}$ and u respectively. The position of the point P is calculated by:

$$Z(P) = \frac{a_1 Z(R) - a_2 Z(Q) + a_1 a_2 [t(R) - t(Q)]}{a_1 - a_2} \tag{16}$$

This procedure is repeated for all nodes and then values of the future output variables are available and one can use common NMPC algorithm to compute the control action. The control action is calculated by solving the following optimization problem in receding horizon manner.

$$\min \int_t^{t+H_p} (T - T_{sp})^2 dt + \int_t^{t+H_c} \lambda (\Delta u)^2 \tag{17}$$

Where H_p is the prediction horizon, H_c is the control horizon, and λ is the weight of the input in the objective function. These parameters are tuning parameters for MPC.

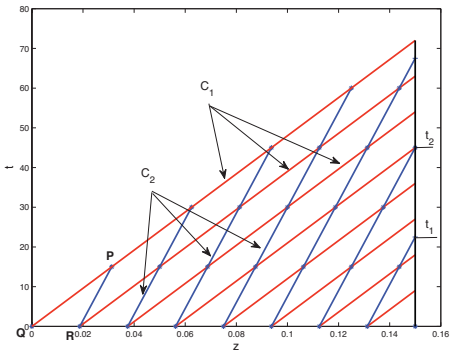


Fig. 2. Calculation of future outputs using characteristic curves

The values of T in the objective function are calculated using the method that described in this section. Since f_1 and f_2 in equations (13) and (14) are nonlinear functions, this optimization problem is a nonlinear optimization and can be solved numerically.

4. NUMERICAL SIMULATIONS

Our case study is a naphtha hydrotreating reactor. The simulation of the reactor was performed using COMSOL® Multi-physics. Using the MPC controller formulated in section 3, the control of the outlet temperature can be achieved. The manipulated variable is the superficial velocity of the feed.

To simulate the closed loop behavior of the system, we performed two cases. In the first one, we considered the deactivation equation of the catalyst and applied the MPC controller to time-varying equations.

In the second case, the model of the system that is used for MPC ignores the catalyst deactivation and assumes constant activity over operation time of the reactor.

Model parameters are given in Table 1. The objective is to control the reactor's outlet temperature at specified setpoint. The objective function is given in Equation (17).

The characteristic curves (9) and (10) are functions of the input variable. Then for the cases that the control horizon, H_c, is greater than one, the characteristic curves will not have constant slope and the calculation of the future values of the output variable will be challenging. In order to simplify the calculations, for the purpose of this example, we assumed that the control horizon is equal to 1, so the MPC problem becomes the following optimization problem:

$$\min \int_t^{t+H_p} (T - T_{sp})^2 dt + \lambda(\Delta u)^2 \tag{18}$$

$$\frac{\partial C_A}{\partial t} = -\frac{\rho_B}{\epsilon} k(t) e^{-\frac{E}{RT}} C_A^{n_1} C_H^{n_2} \quad \text{along characteristic } C_1$$

$$\frac{\partial T}{\partial t} = \frac{\rho_B \Delta H_r}{\rho C_p} k(t) e^{-\frac{E}{RT}} C_A^{n_1} C_H^{n_2} \quad \text{along characteristic } C_2$$

The number of discretization points was taken to be $m = 9$. The prediction horizon is set to the residence time of the reactor, and λ is 1×10^3. The difference between the two cases is in the characteristic equations (11) and (12), which for the first case are functions of time.

This optimization problem can be solved by any optimization method for differential algebraic equations (DAE). Here we used sequential approach, which assumes piecewise constant inputs at each time interval and integrates the differential equations in each interval. This method is an easy method for solving optimization problems for MPC, but it is slower than other algorithms such as that proposed by Bock et al. (2000). The sequential algorithm is good enough for purpose of this illustration example, but for actual implementation the optimization algorithm should be improved.

Fig. 3 illustrates the performance of the CBMPC for the first and second case. This figure shows that the performance of the standard MPC algorithm for the first case is better than the second one. The second case, which considers a constant activity for the catalyst results in an steady state offset. Fig. 5 is the plot of the outlet concentration for two cases and Fig. 4 illustrates the computed control actions for two cases. As Fig. 4 shows, for the second case the input trajectory is almost constant except for first few time intervals; For the first case, due to inclusion of the time varying catalyst activity, the MPC provides more accurate control. Since we assumed

Table 1. Model Parameters

Parameter	Values	unit
ϵ	0.4	
ρ_B	700	$\mathrm{kg_{cat}/m^3}$
C_H	587.4437	$\mathrm{mol/m^3}$
n_1	1.12	
n_2	0.85	
E	81000	J/mol
R	8.314	J/mol K
C_{A0}	0.419344	$\mathrm{mol/m^3}$
C_{Ain}	0.419344	$\mathrm{mol/m^3}$
T_0	523	K
T_{in}	523	K
ρ	2.7	$\mathrm{Kg/m^3}$
C_p	147.49	J/Kg K
ΔH	101.3×10^3	J/mol
α	0.005	
k_1	1.2384	
k_2	2.8896	

piecewise constant profiles for input variable, resulting output trajectory for the first case is non-smooth. But the fluctuations are not greater than $\pm 0.01 \times Y_{sp}$.

In order to deal with the steady state offset problem in the second case, one should implement offset elimination algorithms on standard MPC. These algorithms increase the computational demand of the MPC. Moreover the best offset elimination algorithm may achieve a performance similar to that of the first case.

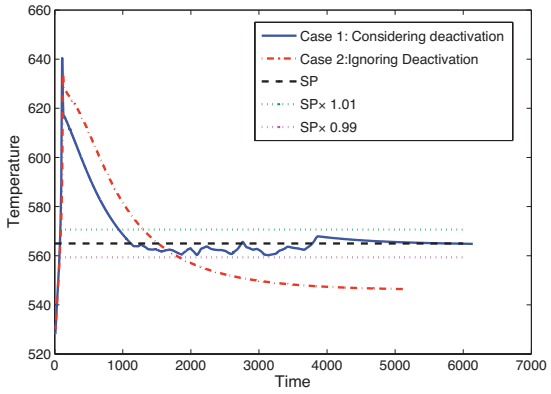


Fig. 3. Outlet temperature(Controlled variable)

Fig. 6 compares the conversion of the reactor for two cases. Although the conversion of the second case is higher at beginning, after a while the reactor's conversion decreases. Lower conversion results in decrease in the profitability of the plant.

5. CONCLUSION

In this work we studied the model predictive control of a naphtha hydrotreating reactor with catalyst deactivation. A characteristic-based MPC is developed to control the reactor. Two different case studies are studied: One that

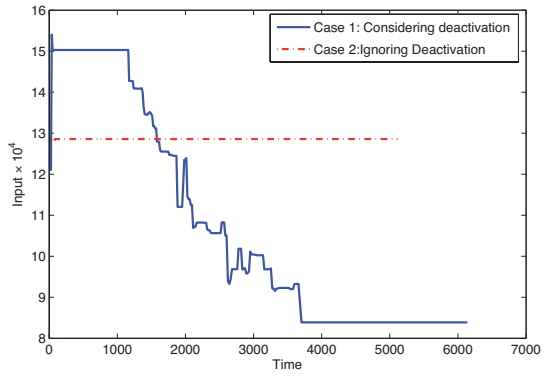


Fig. 4. Computed Input variable

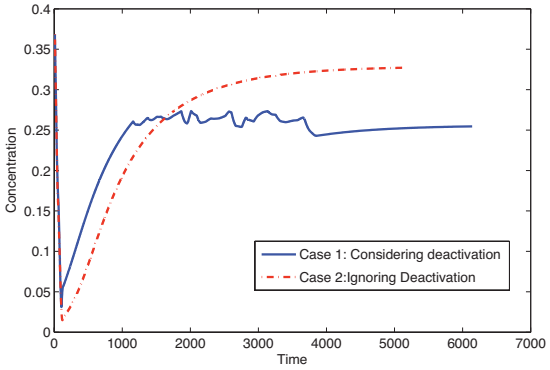


Fig. 5. Computed Input variable

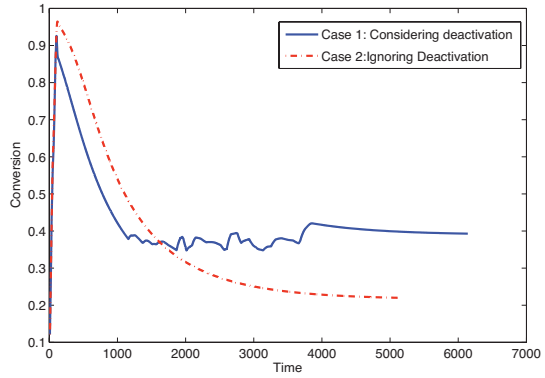


Fig. 6. Conversion at reactor outlet

incorporates the catalyst deactivation kinetics in the controller model and the second one that ignores the catalyst deactivation. The performance of two controllers are compared. The key result of this study is that integration of the catalyst deactivation kinetics with the reactor model, provides improved performance of the characteristic based MPC. This improvement in temperature control results in an improvement in conversion of the reaction, which may increase the plant profitability.

REFERENCES

Aksikas, I., Fuxman, A., Forbes, J.F., and Winkin, J.J. (2009). LQ-Control Design of a Class of Hyperbolic PDE Systems: Application to Fixed-Bed Reactor. *Automatica, in Press.*

Aksikas, I., Fuxman, A.M., and Forbes, J.F. (2008). Control of Time Varying Distributed Parameter Plug Flow Reactor by LQR. In *Proceedings of the IFAC World Congress, Seoul Korea.*

Bock, H., Diehl, M., Leineweber, D., and Scholder, J. (2000). *A direct multiple shooting method for real time optimization of nonlinear DAE processes.* Nonlinear Predictive control. Birkhäuser.

Chen, J., Ring, Z., and Dabros, T. (2001). Modeling and simulation of a fixed-bed pilot-plant hydrotreater. *Ind. Eng. Chem. Res.*, 40, 3294–3300.

Christofides, P.D. (2001). *Nonlinear and Robust Control of PDE Systems:Methods and Applications to Transport-Reaction Processes.* Birkhäuser.

Dubljevic, S., Mhaskar, P., El-Farra, N.H., and Christofides, P. (2005). Predictive control of transport-reaction processes. *Computers & Chemical Engineering*, 29, 2335–2345.

Furimsky, E. and Massoth, F.E. (1999). Deactivation of hydroprocessing catalysts. *Catalysis Today*, 52, 381–495.

Mohammadi, L., Aksikas, I., and Forbes, J.F. (2009). Optimal Control of a Time-Varying Catalytic Fixed Bed Reactor With Catalyst Deactivation. *American Control Conference 2009.*

Shang, H., Forbes, J.F., and Guay, M. (2004). Model predictive control for quasilinear hyperbolic distributed parameter systems. *Ind. Eng. Chem. Res.*, 43, 2140–2149.

Shang, H., Forbes, J.F., and Guay, M. (2007). Computationally efficient model predictive control for convection dominated parabolic systems. *Journal of Process Control*, 17, 379–386.

Hierarchical Economic Optimization of Oil Production from Petroleum Reservoirs

Gijs M. van Essen * **Paul M.J. Van den Hof** *
Jan Dirk Jansen **

* *Delft Center for Systems & Control, Delft University of Technology, Mekelweg 2, 2628 CD Delft, the Netherlands, (e-mail: g.m.vanessen@tudelft.nl, p.m.j.vandenhof@tudelft.nl).*
** *Department of Geotechnology, Delft University of Technology, Stevinweg 1, 2628 CN Delft, the Netherlands / Shell International E&P, Kesslerpark 1 ,2288 GS Rijswijk, the Netherlands (e-mail: jan-dirk.jansen@shell.com).*

Abstract: In oil production *waterflooding* is a popular recovery technology, which involves the injection of water into an oil reservoir. Studies on model-based dynamic optimization of waterflooding strategies have demonstrated that there is a significant potential to increase life-cycle performance, measured in Net Present Value. However, in these studies the complementary desire of oil companies to maximize daily production is generally neglected. To resolve this, a hierarchical optimization structure is proposed that regards economic life-cycle performance as primary objective and daily production as secondary objective. The existence of redundant degrees of freedom allows for the optimization of the secondary objective without compromising optimality of the primary objective.

Keywords: Optimal control, hierarchical structures, redundant DOF, numerical simulation, oil recovery, waterflooding.

1. INTRODUCTION

Oil is produced from subsurface reservoirs. In these reservoirs the oil is contained in the interconnected pores of the reservoir rock under high pressure and temperature. The depletion process of a reservoir generally consists of two production stages. In the primary production stage the reservoir pressure is the driving mechanism for the production. During this phase, the reservoir pressure drops and production gradually decreases. In the secondary production stage liquid (or gas) is injected into the reservoir using injection wells. The most common secondary recovery mechanism involves the injection of water and is referred to as *waterflooding*. It serves two purposes: sustaining reservoir pressure and sweeping the oil out of pores of the reservoir rock and replacing it by water.

Due to heterogeneity of the reservoir rock, the flowing fluids do not experience the same resistance at different points and in different directions in the reservoir. As a result, the oil-water front may not move uniformly towards the production wells, but has a rather irregular shape as depicted schematically in Figure 1. Due to this phenomenon - referred to as *fingering* - the oil-water front may reach the production wells while certain parts of the reservoir are not be properly drained. The produced water must be disposed of in an environmentally friendly way, bringing along additional production costs. At some point the production is no longer economically viable and the wells are closed (shut-in). At the end of the life of the reservoir all production wells are shut-in, while large amounts of oil may still be present in the reservoir.

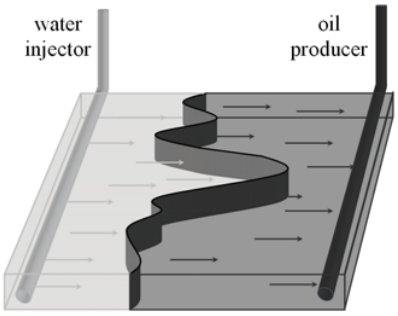


Fig. 1. *Process of waterflooding using a (horizontal) injection and production well. The irregular-shaped oil-water front is a result of the heterogeneous nature of the reservoir, after Brouwer and Jansen (2004).*

Although the injection and production rates of the wells can be manipulated dynamically, they are generally held constant at the maximum capacity of the wells until they are shut-in. Replacing this reactive waterflooding strategy by a dynamic, more proactive one can vastly improve sweep efficiency. Different optimization studies have demonstrated using a numerical reservoir model that there is a potential increase possible of up to 15%, see Brouwer and Jansen (2004) and Jansen et al. (2008). In these optimization studies the objective function is usually of an economic type, most often Net Present Value (NPV), evaluated over the life of the reservoir.

Although many oil companies acknowledge the need for improving economic efficiency over the entire life of the

waterflooding project, many of them adopt maximizing daily production as objective, due to the uncertainty in future economic circumstances. These two objectives, the long-term (life-cycle) objective and the short-term (daily) objective, lead to different, generally conflicting waterflooding strategies.

The goal of this paper is to address the problem of multiple objectives in the optimization of oil recovery from a petroleum reservoir. To that end, a hierarchical optimization structure is proposed that requires a prioritization of the objectives.

This paper proceeds as follows. In Section 2 the properties and characteristics of the reservoir model are described. In Section 3 the life-cycle optimization problem is presented and a hierarchical optimization procedure is proposed. Section 4 deals with identifying redundant degrees of freedom in the optimization problem. The hierarchical optimization procedure is applied to a 3D reservoir model in Section 5. Finally, in Section 6 the results are discussed and alternative approaches are proposed.

2. RESERVOIR MODELING

Reservoir simulators use conservation of mass and momentum equations to describe the flow of oil, water or gas through the reservoir rock. For simplicity reasons, in the oil reservoirs models used within this work only the oil and water phase are assumed to be present.

The mass balance is expressed as follows:

$$\nabla\left(\rho_i u_i\right) + \frac{\partial}{\partial t}\left(\phi \rho_i S_i\right) = 0, \quad i = o, w, \tag{1}$$

where t is time, ∇ the divergence operator, ϕ is the porosity (volume fraction of void space), ρ_i is the density of the phase i, u_i the superficial velocity and S_i the saturation, defined as the proportion of the pore space occupied by phase i.

Conservation of momentum is governed by the Navier-Stokes equations, but is normally simplified for low velocity flow through porous materials, to be described by the semi-empirical Darcy's equation as follows:

$$u_i = -k\frac{k_{ri}}{\mu_i}\nabla p_i, \quad i = o, w, \tag{2}$$

where p_i is the pressure of phase i, k is the absolute permeability, k_{ri} is the relative permeability and μ_i is the viscosity of phase i. The permeability k is an inverse measure of the resistance a fluid experiences flowing through the porous medium. The relative permeability k_{ri} relates to the additional resistance phase i experiences when other phases are present, due to differences in viscosity. As a result, it is a strongly non-linear function of the saturation S_i. In (2) gravity is discarded for simplicity reasons. However, within the 3D example presented in this paper, gravity does play a role. For a more complete description of Darcy's equation we refer to literature, see Aziz and Settari (1979).

Substituting (2) into (1) results into 2 flow equations with 4 unknowns, p_o, p_w, S_o and S_w. Two additional equations are required to complete the system description. The first is the closure equation requiring that the sum of phase saturations must equal 1:

$$S_o + S_w = 1 \tag{3}$$

Second, the relation between the individual phase pressures is given by the capillary pressure equation:

$$p_{cow}(S_w) = p_o - p_w \tag{4}$$

Common practice in reservoir simulation is to substitute (3) and (4) into the flow equations, by taking the oil pressure p_o and water saturation S_w as primary state variables:

$$\nabla(\tilde{\lambda}_o \nabla p_o) = \frac{\partial}{\partial t}\left(\phi\rho_o \cdot [1 - S_w]\right), \tag{5}$$

$$\nabla\left(\tilde{\lambda}_w \nabla p_o - \tilde{\lambda}_w \frac{\partial p_{cow}}{\partial S_w}\nabla S_w\right) = \frac{\partial}{\partial t}\left(\phi\rho_w S_w\right), \tag{6}$$

where $\tilde{\lambda}_o = k\frac{k_{ro}}{\mu_o}$ and $\tilde{\lambda}_w = k\frac{k_{rw}}{\mu_w}$ are the oil and water mobilities. Flow equations (5) and (6) are defined over the entire volume of the reservoir. It is assumed that there is no flow across the boundaries of the reservoir geometry over which (5)-(6) is defined (Neumann boundary conditions).

Due to the complex nature of oil reservoirs, (5)-(6) generally cannot be solved analytically, hence they are evaluated numerically. To this purpose the equations are discretized in space and time. The discretization in space leads to a system built up of a finite number of blocks, referred to as *grid blocks*. This results in the following state space form:

$$\mathbf{V}(\mathbf{x}_k)\cdot\mathbf{x}_{k+1} = \mathbf{T}(\mathbf{x}_k)\cdot\mathbf{x}_k + \mathbf{q}_k, \quad \mathbf{x}_0 = \bar{\mathbf{x}}_0, \tag{7}$$

where k is the time index and $\mathbf{x}$ is the state vector containing the oil pressures (p_o) and water saturations (S_w) in all grid blocks. Vector $\bar{\mathbf{x}}_0$ contains the initial conditions, which are assumed to be known. In the discretization of (5)-(6), the units are converted from $[\frac{kg}{m^3 s}]$ to $[\frac{m^3}{s}]$. In (7) a source vector $\mathbf{q}_k$ is added to model the influence of the wells on the dynamic behavior of the reservoir. The source terms are usually represented by a so-called well model, which relates the source term to the pressure difference between the well and grid block pressure:

$$q_k^j = w^j \cdot (p_{bh,\,k}^j - p_k^j), \tag{8}$$

where $p_{bh,\,k}$ is the well's bottom hole pressure, j the index of the grid block containing the well and p_k^j the grid block pressure in which the well is located. The term w is a well constant which contains the well's geometric factors and the rock and fluid properties of the reservoir directly around the well.

The geological properties inside each grid block are assumed to be constant. The strongly heterogeneous nature of the reservoir can be characterized by assigning different property values to each of the grid blocks. Usually a very large number of grid-blocks is required ($10^3 - 10^6$) to adequately describe the fluid dynamics of a real petroleum reservoir.

The reservoir simulations used within this study are performed using the reservoir simulation software package MoReS, which has been developed by Shell.

3. WATERFLOODING OPTIMIZATION PROBLEM

Flooding a reservoir with water to increase oil production is essentially a batch process, with the additional characteristic that there is no repetition involved. Due to the fact that performance is evaluated at the end of the process and the time constants associated with the nonlinear dynamics are very long, a receding horizon approach will most likely not result in optimal depletion of a reservoir. Dynamic optimization over the entire life of the reservoir is required which can be expressed by the following mathematical formulation:

$$\max_{\mathbf{u}} J(\mathbf{u}), \tag{9}$$

$$s.t.\ \mathbf{x}_{k+1} = \mathbf{f}\left(\mathbf{x}_k, \mathbf{u}_k\right),\ k = 1, .., K, \quad \mathbf{x}_0 = \bar{\mathbf{x}}_0, \tag{10}$$

$$\mathbf{g}(\mathbf{u}) \leq 0 \tag{11}$$

where $\mathbf{u}$ is the input trajectory, $\mathbf{f}$ represents the system equations as described in (7) and $\bar{\mathbf{x}}_0$ is a vector containing the initial conditions of the reservoir. The inequality constraints $\mathbf{g}(\mathbf{u})$ relate to the capacity limitations of the wells.

The objective function J is of an economic type, generally Net Present Value:

$$J = \sum_{k=1}^{K} \left[\frac{r_o \cdot q_{o,k} - r_w \cdot q_{w,k} - r_{inj} \cdot q_{inj,k}}{(1+b)^{\frac{t_k}{\tau_t}}} \cdot \Delta t_k \right], \tag{12}$$

where r_o is the oil revenue $[\frac{\$}{m^3}]$, r_w the water production costs $[\frac{\$}{m^3}]$ and r_{inj} the water injection costs $[\frac{\$}{m^3}]$, which are all assumed constant. K represents the total number of time steps k of a fixed time span and Δt_k the time interval of time step k in $[day]$. The term b represents the discount rate for a certain reference time τ_t. The terms $q_{o,k}$, $q_{w,k}$ and $q_{inj,k}$ represent the total flow rate of respectively produced oil, produced water and injected water at time step k in $[\frac{m^3}{day}]$. An economic objective functions like (12) does not necessarily provide a unique solution to the optimization problem. Although it relates to realistic business conditions, it may well cause ill-posedness of the problem.

Several methods are available for dynamic optimization of large scale problems, see Bryson (1999), Schlegel et al. (2005) and Biegler (2007). *Simultaneous* methods have attractive convergence and constraint handling properties, but even though their capacity to cope with large-scale problems has increased considerably over the recent years, models of order 10^6 still remain very difficult to handle. Although *sequential* methods require repeated numerical integration of the model equations, only the control vector is parameterized and as a result can deal with larger problems. Secondly, due to the fact that the flooding process is very slow much time is available to perform the usually large number of required simulations. However, if the number of control parameters grows the required simulation time may still become unfeasible at some point, unless a method is available to efficiently calculate the gradients of the objective function with respect to the control parameters. This can be done by integration of the adjoint equations or directly through sensitivity equations of model equations.

In the reservoir simulation package used within this work, the adjoint equations are implemented to calculate the gradients. For simplicity reasons, a Steepest Ascent (SA) algorithm is adopted to determine improving control parameters.

3.1 Hierarchical optimization

In the life-cycle waterflooding problem as expressed by (9)-(11), the desire of many oil companies to maximize short-term (daily) production is discarded. A balanced objective provides a possibility to address both objectives in a single function. However, finding an suitable weighting between the objectives may prove to be difficult. Alternatively, we propose a hierarchical (or lexicographic) optimization structure that requires a prioritization of the multiple objectives, as described in Haimes and Li (1988) and Miettinen (1999). In this structure, optimization of a secondary objective function J_2 is constrained by the requirement of the primary objective function J_1 to remain close to its optimal value J_1^*. This structure can be expressed mathematically as follows:

$$\max_{\mathbf{u}} J_2(\mathbf{u}), \tag{13}$$

$$s.t.\ \mathbf{x}_{k+1} = \mathbf{f}\left(\mathbf{x}_k, \mathbf{u}_k\right),\ k = 1, .., K, \quad \mathbf{x}_0 = \bar{\mathbf{x}}_0 \tag{14}$$

$$\mathbf{g}(\mathbf{u}) \leq 0 \tag{15}$$

$$J_1^* - J_1(\mathbf{u}, \mathbf{x}) \leq \varepsilon \tag{16}$$

where ε is an arbitrary small value compared to J_1^*. Solving (13) - (16) requires the knowledge of J_1^*, which is obtained through solving optimization problem (9) - (11).

4. REDUNDANT DEGREES OF FREEDOM

In Jansen et al. (2009) it was observed that significantly different optimized waterflooding strategies result in nearly equal values in NPV. They concluded that the flooding optimization problem is ill-posed and contains many more control variables than necessary. This suggests that optimality of an economic life-cycle objective in waterflooding optimization does not fix all degrees of freedom (DOF) of the decision variable space $\mathcal{D}$, i.e. there exist redundant DOF in the optimization problem. Huesman et al. (2008) found similar results for economic dynamic optimization of plant-wide operation.

A consequence of these redundant DOF is that even if ε in (16) is chosen equal to 0, DOF are left to improve the secondary objective function J_2. A straightforward way of investigating this is to imbed (16) as an equality constraint in the adjoint formulation by means of an additional Lagrange multiplier. Unfortunately, the adjoint functionality in MoReS is not yet capable of dealing with (additional) state constraints. Alternatively, unconstrained gradient information can be used to investigate the redundant DOF, as described in the next section.

4.1 Quadratic approximation of the objective function

A solution $\mathbf{u}$ for which no constraints are active is an optimal solution $\mathbf{u}^*$ if and only if the gradients of J with respect to $\mathbf{u}$ are zero, i.e. $\frac{\partial J}{\partial \mathbf{u}} = 0$. As a result, at $\mathbf{u}^*$ the

gradients do not provide any information on possible redundant degrees of freedom under the optimality condition on J.

Second-order derivatives of J with respect to $\mathbf{u}$ are collected in the Hessian matrix $\mathbf{H}$. If $\mathbf{H}$ is negative-definite, the considered solution $\mathbf{u}$ is an optimal solution, but no DOF are left when the optimality condition on J holds. If $\mathbf{H}$ is negative-semidefinite it means that the Hessian does not have full rank. An orthonormal basis $\mathbf{B}$ for the indetermined directions of $\mathbf{H}$ can than be obtained through a singular value decomposition:

$$\mathbf{H} = \mathbf{U} \cdot \mathbf{\Sigma} \cdot \mathbf{V} \tag{17}$$

The orthonormal basis $\mathbf{B}$ consists of those columns of $\mathbf{V}$ that relate to singular values of zero, i.e. $\mathbf{B} = \{\mathbf{v}_i \mid \sigma_i = 0, \quad i = 1, \ldots, N_{\mathbf{u}}\}$, where $N_{\mathbf{u}}$ is the number of parameters that represent the DOF in the input.

Not all orthogonal directions spanned by the columns of $\mathbf{B}$ will be redundant DOF. These directions are redundant DOF, if they are linear and all higher order derivatives are zero as well, which at this point in time is impossible to proof for reservoir models. $\mathbf{B}$ is however a basis for redundant DOF for a quadratic approximation $\hat{J}$ of objective function J. As $\hat{J}$ can be considered to be an acceptable approximation for small deviations from $\mathbf{u}^*$, $\mathbf{B}$ can be regarded as an acceptable basis for the redundant DOF for small deviations from $\mathbf{u}^*$.

Approximate Hessian matrix Unfortunately, no reservoir simulation package is currently capable of calculating second-order derivatives. However, using the gradient information second-order derivatives can be approximated. Within this work a forward-difference scheme is adopted:

$$\frac{\partial^2 J}{\partial u_i \partial u_j} \approx \frac{\nabla J_i(\mathbf{u} + h_j \mathbf{e}_j) - \nabla J_i(\mathbf{u})}{2h_j} + \frac{\nabla J_j(\mathbf{u} + h_i \mathbf{e}_i) - \nabla J_j(\mathbf{u})}{2h_i} \tag{18}$$

Where $\mathbf{e}_i$ is a canonical unit vector, i.e. a vector with a 1 at element i and 0 elsewhere and h_i is the perturbation step size that relates to parameter u_i of $\mathbf{u}$. In total $N_{\mathbf{u}} + 1$ simulations (function evaluations) are required to obtain the approximate Hessian matrix $\hat{\mathbf{H}}$ at a particular optimal solution $\mathbf{u}^*$.

4.2 Hierarchical optimization method

Adopting the approximation of $\mathbf{H}$ as described in Subsection 4.1, the following iterative procedure is proposed to attack the hierarchical optimization problem (13) - (16) with $\varepsilon = 0$:

(1) Find a (single) optimal strategy $\mathbf{u}^*$ to primary objective function J_1 and use $\mathbf{u} = \mathbf{u}^*$ as starting point in the secondary optimization problem.
(2) Approximate the Hessian matrix $\mathbf{H}$ of J_1 with respect to the input variables at (initial input) $\mathbf{u}$ and determine an orthonormal basis $\mathbf{B}$ for the null-space of $\hat{\mathbf{H}}$.
(3) Find the improving gradient direction $\frac{\partial J_2}{\partial \mathbf{u}}$ for the secondary objective function J_2.

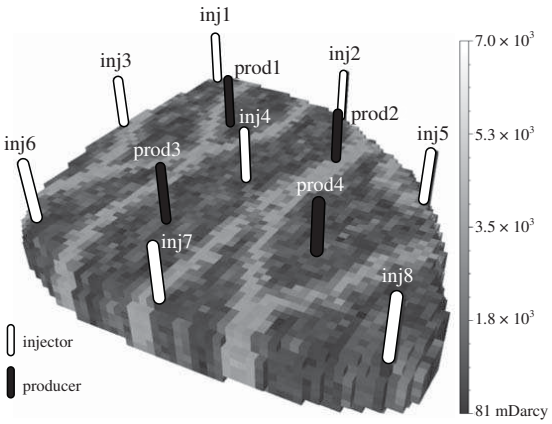


Fig. 2. *3D reservoir model with 4 production and 8 injection wells. The geological structure involves a network of meandering channels in which the fluids flows experience less resistance, due to higher permeability.*

(4) Project $\frac{\partial J_2}{\partial \mathbf{u}}$ onto the orthonormal basis $\mathbf{B}$ to obtain projected direction $\mathbf{d}$, such that $\mathbf{d}$ is an improving direction for J_2, but does not affect J_1. The projection is performed using projection matrix $\mathbf{P}$, see Luenberger (1984):

$$\mathbf{d} = \mathbf{P} \cdot \left(\frac{\partial J_2}{\partial \mathbf{u}} \right)^T \tag{19}$$

$$\mathbf{P} = \mathbf{B} \cdot \left(\mathbf{B} \cdot^T \mathbf{B} \right)^{-1} \cdot \mathbf{B}^T \tag{20}$$

(5) Update $\mathbf{u}$ using projected direction $\mathbf{d}$ in a SA method.

$$\mathbf{u}_{new} = \mathbf{u}_{old} + \tau \cdot \mathbf{d}, \tag{21}$$

where τ is an appropriately small step size such that the quadratic approximation of J_1 is justified.
(6) Perform steps 2 through 6 until convergence of J_2.

In the next section a numerical example is presented where the iterative hierarchical optimization structure is tested on a 3D heterogeneous reservoir model.

5. NUMERICAL EXAMPLE

The hierarchical optimization procedure is applied to a 3-dimensional oil reservoir model, introduced in Van Essen et al. (2006). The life-cycle of the reservoir covers a period of 3,600 days and is chosen such that all oil can be produced within that time frame. The length of the life-cycle is in this example not incorporated as additional optimization parameter. The reservoir model consists of 18,553 grid blocks, as depicted in Figure 2, and has dimensions of $480 \times 480 \times 28$ meter. Its geological structure involves a network of fossilized meandering channels in which the flowing fluids experience less resistance, due to higher permeability. The average reservoir pressure is 400 [*bar*].

The reservoir model contains 8 injection wells and 4 production wells. The production wells are modeled using a well model (8) and operate at a constant bottom hole pressure p_{bh} of 395 [*bar*]. The flow rates of the injection wells can be manipulated directly, i.e. the control input $\mathbf{u}$ involves injection flow rate trajectories for each of the 8

injection wells. The minimum rate for each injection well is 0.0 $[\frac{m^3}{day}]$, the maximum rate is set at a rate of 79.5 $[\frac{m^3}{day}]$.

The control input $\mathbf{u}$ is re-parameterized in time using a zero-order-hold scheme with input parameter vector θ. For each of the 8 injection wells, the control input $\mathbf{u}$ is re-parameterized into 4 time periods t_{θ_i} of 900 days over which the injection rate is held constant at value θ_i. Thus, the input parameter vector θ consists of $8 \times 4 = 32$ elements.

5.1 Life-cycle optimization

The objective function for the life-cycle optimization is defined in terms of NPV, as defined in Equation (12), with $r_o = 126\,[\frac{\$}{m^3}]$, $r_w = 19\,[\frac{\$}{m^3}]$ and $r_i = 6\,[\frac{\$}{m^3}]$. The discount rate b is set to 0. Thus, the life-cycle objective relates to undiscounted cash flow.

The optimal input - denoted by $\mathbf{u}_\theta{}^*$ - obtained after approximately 50 iterations, is shown in Figure 3. None of the input constraints (11) are active for $\mathbf{u}_\theta{}^*$. The value of the objective function corresponding to input $\mathbf{u}_\theta{}^*$ is 47.6×10^6 \$.

5.2 Hierarchical optimization

A secondary objective function J_2 was defined to emphasize the importance of short-term production. To that end, J_2 is chosen identical to the primary objective function but with the addition of a very high annual discount rate b of 0.25. As a result, short-term production is weighed far more heavily than future production. Note that due to the very high discount rate, the actual value of J_2 no longer has a realistic meaning in an economic sense.

The hierarchical approach as presented in Subsection 4.2 is applied. The total number of simulation runs needed to approximate the Hessian ($\hat{\mathbf{H}}$) is 33. However, the required simulation time was vastly reduced by parallel processing the simulations.

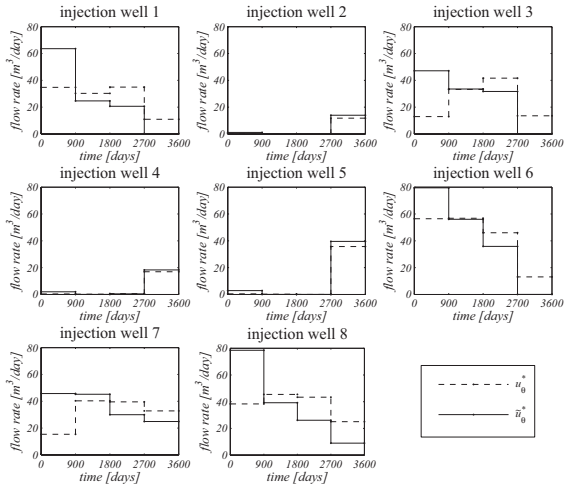


Fig. 3. *Input trajectories for each of the 8 injection wells for the initial optimal solution $\mathbf{u}_\theta{}^*$ to J_1 (dashed line) and the optimal solution $\tilde{\mathbf{u}}_\theta^*$ after the constrained optimization of J_2 (solid line)*

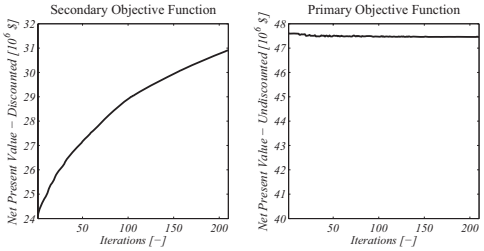


Fig. 4. *Values of the secondary J_2 and primary J_1 objective function plotted against the iteration number for the constrained secondary optimization problem.*

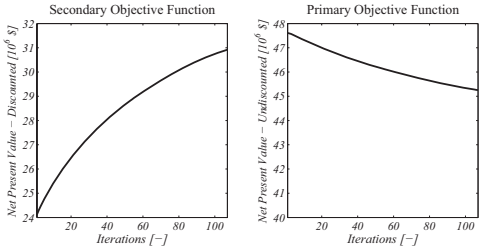


Fig. 5. *Values of the secondary J_2 and primary J_1 objective function plotted against the iteration number for the secondary optimization problem, no longer constrained by the orthonormal basis $\mathbf{B}$.*

Due to the fact that this example involves a numerical model and an approximation of the second-order derivatives, the selection criterion for $\mathbf{B}$ is relaxed. Those columns $\mathbf{v}_i$ of $\mathbf{V}$ were selected that correspond to singular values for which $\frac{\sigma_i}{\sigma_1} < 0.02$ instead of $\sigma_i = 0$. The projected gradients $\mathbf{d}$ were again used in a steepest-ascent scheme. For the quadratic approximation of J_1 to be justified, $\mathbf{u}_{\theta,new}$ must remain close to $\mathbf{u}_{\theta,old}$. To achieve that, $\mathbf{d}$ was normalized and a constant step size τ of 1 was used. Due to time restrictions, the hierarchical optimization of J_2 was terminated after 210 iterations with final control input $\tilde{\mathbf{u}}_\theta^*$. To evaluate the results of the hierarchical optimization, a second optimization case was carried out, where optimization of J_2 was performed *without* projection on $\mathbf{B}$. As a result, the obtained control input - denoted by $\tilde{\mathbf{u}}_\theta$ - does in this case not ensure optimality of J_1.

Figure 4 displays the values of J_1 and J_2 plotted against the iteration number for the hierarchical optimization problem. It shows a considerable increase of J_2 of 28.2% and a slight drop of J_1 of -0.3%. In Figure 3 the input strategy after the final iteration step is presented. It can be observed that the injection strategy shows a substantial increase in injection rates at the beginning of the production life and a decrease at the end. The values of J_1 and J_2 plotted against the iteration number for the *unconstrained* optimization of J_2 are shown in Figure 5. Again an increase of J_2 of 28.2% is realized, but now at a cost of a decrease of J_1 of -5.0%. Finally, Figure 6 shows the value of the primary objective function J_1 over time until the end of the producing reservoir life for $\mathbf{u}_\theta{}^*$, $\tilde{\mathbf{u}}_\theta^*$ and $\tilde{\mathbf{u}}_\theta$. Input $\tilde{\mathbf{u}}_\theta^*$ shows a steeper ascent of J_1 than $\mathbf{u}_\theta{}^*$, while their final values are nearly equal. Input $\tilde{\mathbf{u}}_\theta$ shows initially the same steep ascent as $\tilde{\mathbf{u}}_\theta^*$, but J_1 drops at the end of the life of the reservoir.

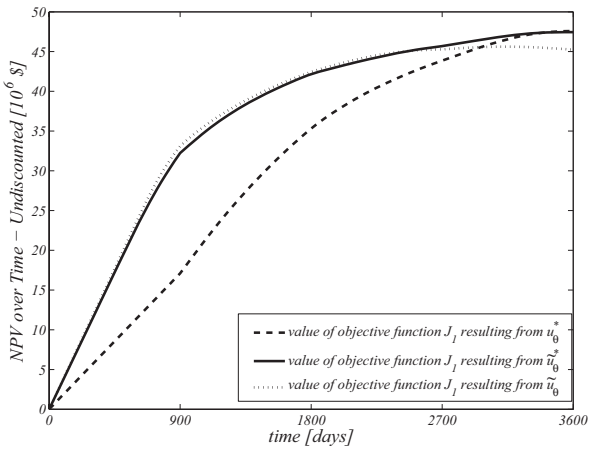


Fig. 6. *Value of the primary objective function J_1 over time for initial optimal input $\mathbf{u}_\theta{}^*$ to J_1 (dashed line), the optimal input $\tilde{\mathbf{u}}_\theta^*$ after the constrained optimization of J_2 (solid line) and input $\tilde{\mathbf{u}}_\theta$ after the unconstrained optimization of J_2 (dotted line)*

6. CONCLUSION

Model-based optimization is a relatively new approach to oil recovery from petroleum reservoirs. Optimization studies have shown a considerable potential increase in life-cycle performance. However, increased understanding of the optimal control problem and characteristics of the optimal solutions is necessary to take the next step towards a real-life application.

Within this work the issue of multiple objectives in oil production is addressed. A hierarchical approach is investigated by means of a simulation experiment. For the presented experiment we conclude that:

- There exist redundant DOF in the input strategy $\mathbf{u}$ with respect to the optimality of the life-cycle objective. This implies the existence of an optimal subset $\mathcal{S}$ of connected optimal solutions within the solution space $\mathcal{D}$.
- The redundant DOF create enough freedom to significantly improve the secondary objective function. Moreover, the difference between the initial and final input strategy to the secondary optimization problem is substantial. This suggest that $\mathcal{S}$ occupies a considerable space within decision variable space $\mathcal{D}$.
- The presented hierarchical optimization procedure provides a method to incorporate short-term performance objectives into problem setting of maximizing life-cycle performance of oil recovery. Using the hierarchical structure, optimization of the secondary objective may be executed without significantly compromising the primary objective.

Under which conditions these conclusions also apply to different life-cycle waterflooding problems and/or different reservoir models will be subject for further investigation.

6.1 Discussion

The presented hierarchical optimization approach is computationally very demanding and becomes infeasible for more realistic reservoir models with an increased number of input parameters. A different method to approximate the Hessian requiring less simulation runs may be considered to resolve this, e.g. the secant method. However, calculating second-order derivatives may be avoided altogether when the hierarchical optimization problem is imbedded in the adjoint formulation, as mentioned in Section 4. This approach will be the focus of future research.

Within this work, uncertainty - of the model and/or the objective function parameters - was neglected. In literature, a number of methods are presented to attack the problem of life-cycle optimization under uncertainty, using a closed-loop approach. For a good overview see Jansen et al. (2008). Without considerable effort, the presented hierarchical optimization structure can be integrated into this closed-loop framework.

REFERENCES

Aziz, K. and Settari, A. (1979). *Petroleum Reservoir Simulation*. Applied Science Publishers.

Biegler, L.T. (2007). An overview of simultaneous strategies for dynamic optimization. *Chemical Engineering and Processing: Process Intensification*, 46(11), 1043–1053. doi:10.1016/j.cep.2006.06.021.

Brouwer, D.R. and Jansen, J.D. (2004). Dynamic optimization of waterflooding with smart wells using optimal control theory. *SPE Journal*, 9(4), 391–402. doi:10.2118/78278-PA. SPE 78278-PA.

Bryson, A.E. (1999). *Dynamic Optimization*. Addison Wesley Longman.

Haimes, Y.Y. and Li, D. (1988). Hierarchical multiobjective analysis for large-scale systems: Review and current status. *Automatica*, 24(1), 53–69. doi:10.1016/0005-1098(88)90007-6.

Huesman, A.E.M., Bosgra, O.H., and Van den Hof, P.M.J. (2008). Integrating mpc and rto in the process industry by economic dynamic lexicographic optimization; an open-loop exploration. In *AIChE Annual Meeting*. Philadelphia, U.S.A.

Jansen, J.D., Bosgra, O.H., and Van den Hof, P.M.J. (2008). Model-based control of multiphase flow in subsurface oil reservoirs. *Journal of Process Control*, 18(9), 846–855. doi:10.1016/j.jprocont.2008.06.011.

Jansen, J.D., Douma, S.D., Brouwer, D.R., Van den Hof, P.M.J., Bosgra, O.H., and Heemink, A.W. (2009). Closed loop reservoir management. In *SPE Reservoir Simulation Symposium*. The Woodlands, Texas, U.S.A. doi:10.2118/119098-MS. SPE 119098-MS.

Luenberger, D.G. (1984). *Linear and nonlinear programming*. Addison-Wesley.

Miettinen, K.M. (1999). *Nonlinear Multiobjective Optimization*. Kluwer Academic Publishers, Boston.

Schlegel, M., Stockmann, K., Binder, T., and Marquardt, W. (2005). Dynamic optimization using adaptive control vector parameterization. *Computers & Chemical Engineering*, 29(8), 1731–1751. doi:10.1016/j.compchemeng.2005.02.036.

Van Essen, G.M., Zandvliet, M.J., Van den Hof, P.M.J., Bosgra, O.H., and Jansen, J.D. (2006). Robust waterflooding optimization of multiple geological scenarios. In *SPE Annual Technical Conference and Exhibition*. San Antonio, Texas, U.S.A. doi:10.2118/102913-MS. SPE 102913-MS.

Expected Cost Optimization using Asymmetric Probability Density functions

Bertrand Pigeon*. Michel Perrier*. Bala Srinivasan*

**NSERC Environmental Design Engineering Chair in Process Integration,*
Department of Chemical Engineering, École Polytechnique de Montréal,
C.P. 6079, Succ. Centre Ville, Montréal, Québec, Canada, H3C 3A7,
(e-mail: bertand.pigeon@ polymtl.ca)

Abstract: In the stochastic context, expected value of the cost function is optimized either by changing the mean values of the manipulated variables or by reducing their variance. An extension is to look for an optimal shape for the entire probability density function (PDF). Though the use of asymmetric PDFs is proposed in the literature, no formal proof that justifies their use has been provided. In this paper, it is shown that an asymmetric PDF is required if and only if the cost function is asymmetric and the manipulated variable is penalised. The proof uses an analytical solution of the Fokker-Planck-Kolmogorov equation derived to calculate the shape the output PDF for scalar systems. In particular, this analytical solution is adapted to a switching proportional controller. The theoretical concepts are illustrated on a simulation example, where the advantage of choosing an asymmetric PDF is shown.

Keywords: Stochastic control, Optimization, Probability Density Functions, Switching Algorithms

1. INTRODUCTION

Optimization in a stochastic context involves studying the influence of decisions variables on the expected value of the objective function. In the stochastic context, not only the mean values of the decision variables but the entire distribution plays a role in optimization. Typically, in the presence of constraints, variability is reduced first using appropriate controllers, and secondly by shifting the set point closer to the constraint. Use of minimum variance controllers for optimization purposes has been well studied in the literature (Muske, 2003).

However, shaping the entire probability density function (PDF) could be a viable option to reduce costs. The first mention of this possibility was made in Kàrnỳ (1996). Then, Wang (1998) developed a PDF shaping algorithm based on the weights of a neural B-Spline that parameterized the output PDF. This method has been improved ever since by the same authors (Wang ,2002; Wang & Zhang 2002; Wang & Wang, 2002; Guo & Wang 2005). Crespo and Sun (2002) used an analytical solution of Fokker-Plank-Kolmogorov equation in steady-state to develop a PDF shaping algorithm. On the other hand, Forbes et al. (2004) developed an algorithm based on the parametrization of the target PDF using Gram-Charlier basis functions. In all the above cited works, though the motivation is to improve an optimization objective, only the sub-problem of getting close to a target PDF is addressed. No indication is given on how to compute a target PDF that is suited for the optimization problem at hand.

It has been argued in all the above works that the advantage of PDF shaping lies in shaping it in an asymmetric manner. The necessity of an asymmetric PDF arises from the asymmetry of the objective function. This is normally due to the presence of process and operational constraints. With constraints, typically, an approach based on penalty (barrier) function is used for resolution. An additional cost is added when the constraint is violated (or in the barrier function case an additional cost is added when operated close to the constraint), which inturn causes asymmetry.

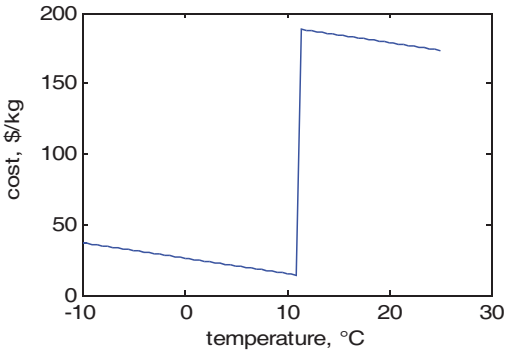


Figure 1: Example of an asymmetric objective function

Figure 1 shows an example with a penalty function where a constant penalty is added if the manipulated variable is above the constraint set at 11°C. As seen, such penalty/barrier functions cause a huge asymmetry around the optimal solution. The optimal solution without any stochastic behaviour would be on the constraint 11°C. However with process noise, a controller needs to be used to reduce the variance of the manipulated variable, and the set point must be lower than 11°C, so that only a small

part of the distribution violates the constraint. The minimum variance controller tries to squeeze and shift the distribution towards the constraint. On the other hand, the PDF shaping solution tries to match the asymmetry in the objective function using an asymmetric PDF with its tail on the opposite side of the constraint.

Though intuitive arguments were given for using asymmetric PDFs, no formal results are available to distinguish the cases where an asymmetric PDF would be more beneficial than the symmetric one. So, the main question asked in this paper is, "which class of problems requires an asymmetric PDF?" It is shown that not only the asymmetry of the objective function but also an input weighting is needed to necessitate an asymmetric PDF. The importance of input weighting is one of the core contributions of this paper. In the minimum variance controller, by reducing the variability of the output variable, the variability of the manipulated variables would increase, straining the process equipment. Contrarily, with an asymmetric PDF, the set point can be shifted toward the constraint and with less impact on the manipulated variables.

This paper first presents an analytical solution of the Fokker-Planck-Kolmogorov (FPK) equation for general scalar systems. This analytical solution is then applied to the switching controller case, using which, the optimality or non-optimality of symmetric solution is ascertained. The last section is devoted to a simulation example where the improvement in cost using an asymmetric controller is shown.

2. PROBLEM FORMULATION

2.1Optimization problem formulation

Consider the dynamic system given by equation (1), where u is the scalar manipulated variable, x the scalar state variable, and w the zero-mean Gaussian process noise input with standard deviation η .

$$\dot{x} = f(x) + g(x)u + w \tag{1}$$

The functions $f(x)$ and $g(x)$ represent the unforced and the forced parts of the system dynamics. Consider the optimization of the above system at steady state:

$$\begin{aligned} &\min_{u} \Phi(x,u) \\ &C(x,u) \leqslant 0 \\ &f(x) + g(x)u = 0 \end{aligned} \quad , \tag{2}$$

where Φ is the function to be optimized, C the constraints. Note that the optimization considers the system equations without noise at steady state as equality constraints.

In the context of this paper, a penalty function is introduced to handle the constraints as show below:

$$\begin{aligned} &\min_{u} [\bar{\Phi}(x,u) + D(C(x,u))] = \phi(x,u) \\ &f(x) + g(x)u = 0 \end{aligned} \quad , \tag{3}$$

where $D(.)$ is any appropriate penalty function and $\phi(.)$ the augmented cost. As discussed earlier, $D(.)$ is asymmetric which would lead to an asymmetry in the cost function.

In the context of this paper, x is considered stochastic due to the presence of the noise term w. So, the expectation of the cost function needs to be calculated for optimization purposes. The cost function that is minimized is given by:

$$J = \int_{-\infty}^{+\infty} \int_{-\infty}^{+\infty} \phi(x,u)\, p(x,u)\, dx du \quad , \tag{4}$$

where $p(x,u)$ is the joint probability density function.

2.3 Controllers for PDF shaping

In this section, the nonlinear controller used for PDF shaping is presented. Nonlinearity is crucial since, if the process and the controller were linear, and the input is Gaussian, the output PDF would just be Gaussian.

In order to have a full control on the nonlinearity, all the system nonlinearities are eliminated by feedback linearization. In addition, the controller $h(x)$ is used to bring the state to its desired set point. Then, the controller would introduce the nonlinearities required to shape the PDFs. For the system under consideration, the linearizing feedback is given by:

$$u = \frac{-f(x) + h(x)}{g(x)} \quad . \tag{5}$$

Here, a switching controller of the following form will be studied. The nonlinearity arises from the gain schedule and results in an asymmetrical PDF.

$$h(x) = \begin{Bmatrix} k_1(x_{sp} - x) & if & x < x_{sp} \\ k_2(x_{sp} - x) & if & x > x_{sp} \end{Bmatrix} \quad . \tag{6}$$

To simplify the development, no measurement noise is considered, while a zero-mean Gaussian measurement noise, z, with standard deviation λ will be added to the set point. Thus, the the system reads

$$\begin{aligned} &\dot{x} = h(x) + w + k_{cont}(x) z \\ &k_{cont}(x) = \begin{Bmatrix} k_1 & if & x < x_{sp} \\ k_2 & if & x > x_{sp} \end{Bmatrix} \end{aligned} \tag{7}$$

3.ANALYTICAL SOLUTION OF THE SCALAR FPK EQUATION FOR SWITCHING CONTROLLER

In this section, the analytical solution of the FPK equation will be developed for the general $h(x)$ and later exploited to suit the switching controller.

3.1 General case

Consider the system (8) where the two random variables are Brownian processes :

$$dx = h(x)dt + \eta\, d\beta_w + k_{cont}(x)\,\lambda\, d\beta_z, \quad t \geq t_0 \quad , \qquad (8)$$

where η and λ are the standard deviations of the process and measurement noise respectively, $d\beta_w$ and $d\beta_z$ are unit variance Brownian processes. These two noises can be clubbed together into a general equation as follows:

$$dx = h(x)dt + \rho(x)\, d\beta, \quad t \geq t_0 \qquad (9)$$

where $\rho(x)$ represents the agglomerated standard deviation.

The evolution of its probability density function of *x* is given by the Fokker-Planck-Kolmogorov equation (Jazwinsky (1968)):

$$\frac{\partial p(x,t)}{\partial t} = \frac{-\partial[p(x,t)h(x)]}{\partial x} + \frac{1}{2}\frac{\partial^2[p(x,t)\rho(x)^2]}{\partial x^2} \quad . (10)$$

with boundary conditions $lim_{x\to\infty} p = lim_{x\to-\infty} p = 0$

$lim_{x\to\infty} \frac{\partial p}{\partial x} = lim_{x\to-\infty} \frac{\partial p}{\partial x} = 0$, $\int_{-\infty}^{\infty} p(x)\,dx = 1$.

At steady state, this equation reads

$$\frac{d\,p\,h}{d\,x} = \frac{1}{2}\frac{d^2\rho^2 p}{d\,x^2} \quad . \qquad (11)$$

By integrating both sides of the equation:

$$2\,ph = \frac{d\,\rho^2 p}{d\,x} + c \quad . \qquad (12)$$

Using the boundary conditions it can be seen that, *c = 0.* Rearranging the terms gives,

$$\frac{d\,p}{p} = \left(\frac{2\,h}{\rho^2} - \frac{2}{\rho}\frac{d\,\rho}{d\,x}\right) d\,x \quad . \qquad (13)$$

The solution of the above equation is given by:

$$p(x) = p_0\, e^{\int_{-\infty}^{\infty}\left(\frac{2h}{\rho^2} - \frac{2}{\rho}\frac{d\rho}{dx}\right)dx} \quad , \qquad (14)$$

where p_0 is the normalizing constant to render the integral of the probability to 1.

3.2 Switching controller case

The analytical solution developed in Section 3.1 is applied to a case of the switching controller. Let p_{sp} be the value of the probability density function at $x = x_{sp}$. From (6) and (7) it can be seen that to the left of the set point

$$h(x) = k_1(x - x_{sp}),\ \rho^2(x) = \eta^2 + k_1^2\lambda^2 \quad , \qquad (15)$$

and to the right

$$h(x) = k_2(x - x_{sp}),\ \rho^2(x) = \eta^2 + k_2^2\lambda^2 \quad . \qquad (16)$$

Thus, it can be seen that

$$p(x) = \begin{cases} p_{sp}\, e^{\frac{-2k_1(x-x_{sp})^2}{\eta^2 + k_1^2\lambda^2}} & \text{for } x < x_{sp} \\ p_{sp}\, e^{\frac{-2k_2(x-x_{sp})^2}{\eta^2 + k_2^2\lambda^2}} & \text{for } x \geq x_{sp} \end{cases} \quad . \qquad (17)$$

This can be interpreted as Gaussian function where the two branches are not symmetric. The variance on one side is different from that of the other. The variances on either side can be computed as follows:

$$\sigma_1 = \frac{\sqrt{\eta^2 + k_1^2\lambda^2}}{2\sqrt{k_1}} \quad \text{and} \quad \sigma_2 = \frac{\sqrt{\eta^2 + k_1^2\lambda^2}}{2\sqrt{k_2}} \quad . \qquad (18)$$

Also, the normalisation constant can be computed analytically as follows:

$$p_{sp} = \frac{2}{\sqrt{2\pi}(\sigma_1 + \sigma_2)} \quad . \qquad (19)$$

From the expression of *h(x)* it can also be shown that

$$p(h) = \begin{cases} p_{h0}\, e^{\frac{-2h^2}{k_1(\eta + k_1\lambda)^2}} & \text{for } h < 0 \\ p_{h0}\, e^{\frac{-2h^2}{k_2(\eta + k_2\lambda)^2}} & \text{for } h \geq 0 \end{cases} \qquad (20)$$

with

$$\sigma_{1h} = \frac{\sqrt{\eta^2 + k_1^2\lambda^2}\sqrt{k_1}}{2} \quad , \quad \sigma_{2h} = \frac{\sqrt{\eta^2 + k_2^2\lambda^2}\sqrt{k_2}}{2} \qquad (21)$$

$$p_{h0} = \frac{2}{\sqrt{2\pi}(\sigma_{1h} + \sigma_{2h})} \quad . \qquad (22)$$

4.NON-OPTIMALITY OF THE SYMMETRIC SOLUTION

In this section, it is be shown that a symmetric PDF is sufficient even for an asymmetric objective function, when there is no input weighting. Also, when the objective function is symmetric, with or without input weighting a symmetric PDF is indeed optimal. However, when there is asymmetry and input weighting, then it is shown that a symmetric solution is not optimal.

Consider equation (4). Since, *u* is a function of *x*, the objective function $\phi(x,u)$ is just a function of *x*. In particular, consider a special case where the squared deviation of the control action *h(x)* is included in the cost function. The remaining part of the objective function is termed *l(x)*. So,

$$\phi(x,u) = l(x) + \gamma\, h^2(x) \qquad (23)$$

Due to the imposed control structure, the degree of freedom for the optimization problem is no longer u, but the parameters x_{sp}, k_1 and k_2. So, the optimization problem reads,

$$\min_{x_{sp},k_1,k_2} J=\int_{-\infty}^{+\infty} l(x)\,p(x)\,dx+\int_{-\infty}^{+\infty}\gamma h^2\,p(h)\,dh \quad . \tag{24}$$

The proof of non-optimality proceeds by deriving the necessary conditions of optimality of the above optimization problem by considering that k_1 and k_2 are varied independently. Then an additional condition of symmetry, i.e. $k_1 = k_2$ is imposed. This gives four conditions (3 necessary conditions and one condition of symmetry) for three variables. If these four conditions are consistent then the symmetric solution is indeed optimal. On the other hand, if it leads to an inconsistency or contradiction then it shows that the symmetric solution is not optimal in the case considered.

Theorem 1: The symmetric switching controller is locally optimal if and only if (i) l(x) is symmetric around the optimum, i.e., the third derivative evaluated at the optimum is zero, or (ii) the input weighting γ is zero.

Proof: Without loss of generality let $x = 0$, $l(x) = 0$, $J = 0$ be the optimum in the absence of noise. Consider the third order Taylor series expansion of $l(x)$ around $x = 0$. The first two terms are zero since $l(0) = 0$ and the first derivative is zero due to optimality. Thus the expansion is given by

$$l(x)=\alpha x^2+\delta x^3 \quad , \tag{25}$$

where α and δ are the second and third derivatives, respectively, at the origin. The expected cost (5) is then given by,

$$J=\alpha\int_{-\infty}^{+\infty} x^2\,p(x)\,dx+\delta\int_{-\infty}^{+\infty} x^3\,p(x)\,dx+\gamma\int_{-\infty}^{+\infty} h^2\,p(h)\,dh \quad .(26)$$

Analytical expressions for all the three terms can be obtained.

$$\int_{-\infty}^{+\infty} x^2\,p\,dx=x_{sp}^2-\frac{4\,x_{sp}(\sigma_1-\sigma_2)}{\sqrt{2\pi}}+(\sigma_1^2-\sigma_1\sigma_2+\sigma_2^2) \tag{27}$$

$$\int_{-\infty}^{+\infty} x^3\,p\,dx=x_{sp}^3-\frac{6\,x_{sp}^2}{\sqrt{2\pi}}(\sigma_2-\sigma_1)+3\,x_{sp}(\sigma_2^2-\sigma_1\sigma_2+\sigma_1^2)\frac{+4}{\sqrt{2\pi}}(\sigma_2-\sigma_1)(\sigma_1^2+\sigma_2^2) \tag{28}$$

$$\int_{-\infty}^{+\infty} h^2\,p(h)\,dh=(\sigma_{h1}^2-\sigma_{h1}\sigma_{h2}+\sigma_{h2}^2)-\frac{2}{\pi}(\sigma_{h1}-\sigma_{h2})^2 \tag{29}$$

The optimality condition requires that the derivatives of J with respect to x_{sp}, k_1 and k_2 be zero, the expressions for which can be readily obtained. To analyse the symmetric solution, consider $k_1 = k_2 = k$. Substituting this in the derivatives leads to

$$\frac{\partial J}{\partial x_{sp}}=2\alpha x_{sp}+3\delta x_{sp}^2+\frac{3\delta}{4k}(\eta^2+k^2\lambda^2)=0 \quad , \tag{30}$$

$$\frac{\partial J}{\partial k_1}-\frac{\partial J}{\partial k_2}=\frac{(\eta^2-k^2\lambda^2)}{\sqrt{2\pi k^3}}\left(2\alpha x_{sp}+3\delta x_{sp}^2+\frac{\delta}{k}(\eta^2+k^2\lambda^2)\right)=0 \quad , \tag{31}$$

and

$$\frac{\partial J}{\partial k_1}+\frac{\partial J}{\partial k_2}=\frac{\eta^2-k^2\lambda^2}{4k^2}(\alpha+3\delta x_{sp})+\frac{\gamma}{4}(\eta^2+3k^2\lambda^2)=0 \quad . \tag{32}$$

It can be seen that there are 3 equations for 2 unknowns, k and x_{sp}. Replacing the terms with x_{sp} in (31) using (30), it can be seen that

$$\frac{\partial J}{\partial k_1}-\frac{\partial J}{\partial k_2}=\frac{(\eta^2-k^2\lambda^2)}{\sqrt{2\pi k^3}}\frac{\delta}{4k}(\eta^2+k^2\lambda^2)=0 \quad . \tag{33}$$

Only if part: $\delta\neq 0, \gamma\neq 0 \Rightarrow non-optimality$

When $\delta\neq 0$, the only solutions of (33) are $k=\pm\eta/\lambda$ But, plugging these values of k in the sum of derivatives lead to $\gamma=0$. So, if $\gamma\neq 0$ the symmetric controller is not optimal .

If part: $\gamma=0\Rightarrow optimality$

When $\gamma=0$ note that $k=\pm\eta/\lambda$ satisfies all the three necessary conditions of optimality .

If part: $\delta=0\Rightarrow optimality$

Since $\delta=0$, (31) gives $x_{sp} = 0$. (33) is not useful in determining k. However from (32), it can be seen that the following 4th order equation can be used to compute k.

$$3\gamma\lambda k^4+(\gamma\eta+\lambda\alpha)k^2-\alpha\eta=0 \tag{34}$$

■

5. EXAMPLE

In this section, an asymmetric example with input weighting is presented. The optimal switching controller is computed using the output PDF obtained through the analytical solution. It will be shown that such a controller indeed leads to an asymmetric PDF.

A cost function analogous to the one in Figure 1 is considered here.

$$\begin{gathered}\phi(x)=26-10x+D(c(x))+10h^2(x)\\ D(c(x))=0 \;\; if \;\; x\leqslant 11\\ D(c(x))=10^5 \;\; if \;\; x>11\end{gathered} \tag{35}$$

The system dynamics is given by

$$\dot{x}=-0.4\mathrm{x}+0.2\mathrm{u}+w \quad , \tag{38}$$

where the process noise w has a mean of 0 and a standard deviation $\eta=1$. A measurement noise of standard deviation $\lambda=0.01$ was considered. Though it is unrealistic to consider a ratio of 100 between the standard deviations of

process and measurement noises, it is required in this case to prove the principle. The asymmetric PDF gives better results only in a narrow range of parameter values and so is such a choice made.

5.1 Controller design

The controller (6) is used here. It has 3 parameters; gains k_1, k_2 and the set point x_{sp}. These parameters are found via non linear programming where the equation (24) is minimized. Equation (24) for the given example can be written as follows:

$$J = 26 - 10\int_{-\infty}^{+\infty} x\,p(x)\,dx + 10^5 \int_{11}^{+\infty} p(x)\,dx + 10\int_{-\infty}^{+\infty} v^2\,p(v)\,dv \tag{39}$$

Also, in this case an analytical expression for all the three terms can be derived using $p(x)$ given in (20). The analytical expression of the last term is already provided in (31). The expressions for the other terms are given as follows:

$$\int_{c}^{+\infty} p(x)\,dx = \frac{\sigma_2}{\sigma_1+\sigma_2}\left(1+\mathrm{erf}\left(\frac{x_{sp}-c}{\sqrt{2}\,\sigma_2}\right)\right) \tag{48}$$

$$\int_{-\infty}^{+\infty} x\,p(x)\,dx = x_{sp} + \sqrt{\frac{2}{\pi}}\left(\sigma_2-\sigma_1\right) \quad . \tag{49}$$

5.2 Results

The optimal parameters for a switching controller and a constant gain control have been found numerically. For calculating the optimal single-gain controller, the same calculations are used with $k_1 = k_2$. The optimal gains and the value of the cost function are presented in Table 1. It can be seen that with the switching controller, the cost is reduced by around 6.7%. It is because by having 2 gains, the controller can be aggressive on one side, the side of the constraint, while having a low gain and thereby low input variance on the other side.

Table 1: Results of the example

Controller type	Switching controller	Single gain controller
Set point	10.6	10.53
k_1	0.41	2.94
k_2	4.99	2.94
Cost	5.39	5.78

Figure 2 shows the output PDF for the both controllers. It can be seen that the single proportional controller leads to a symmetric Gaussian PDF, while with the switching controller results in an asymmetric PDF. It is equally interesting to see in Figure 3 that the asymmetry in the input PDF is reversed.
It can be explained by the fact that closer to the constraint, the input works hard and has a larger variance, while far from the constraint, the input does not work in order to reduce the cost by decreasing its variance.

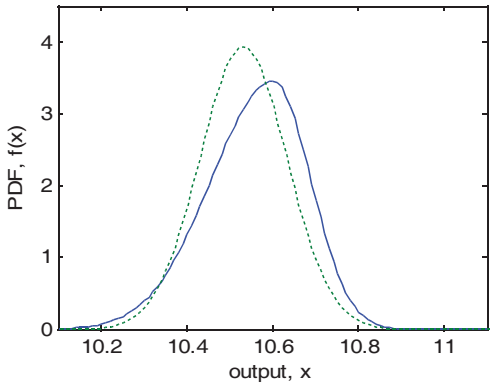


Figure 2: Output PDF with a switching controller (solid line) and with a non-switching controller (dotted line)

Several tests were performed with varying penalties, with varying input weights, and varying measurement noise levels. Figure 4 shows the effect changing the penalty. It can be seen that increasing the weighting for the penalty increases the difference between the cost functions of the symmetric and asymmetric PDF. This tendency can be attributed to the fact that increasing the penalty increases the asymmetry of the cost function. Note that the x axis is logarithmic, i.e, a small increase in the difference calls for a order of magnitude change in the weighting.

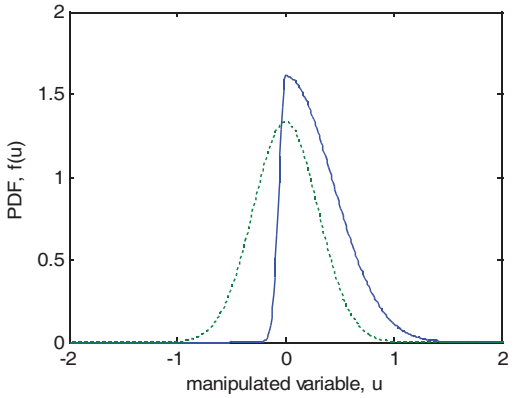


Figure 3: Manipulated variable PDF with a switching controller (solid line) and with a non-switching controller (dotted line)

Figure 5 shows the effect of changing the input weight. An interesting effect can be observed here. The difference first increases, while it decreases after reaching a maximum. Intuitively, when the input weight is zero, the symmetric solution is indeed optimal and there can be no gain by using an asymmetric controller. On the other hand, since the input weighting is symmetric, for large input weightings the asymmetry of the cost function becomes negligible and so a symmetric controller is again optimal.

Figure 6 shows the influence of measurement noise on the difference. The larger the measurement noise, lesser is the gain that can be obtained by using an asymmetric PDF. This is due to the fact that with increasing measurement noise the minimum variance controller as such has a fairly low gain and not much manoeuvrability is left.

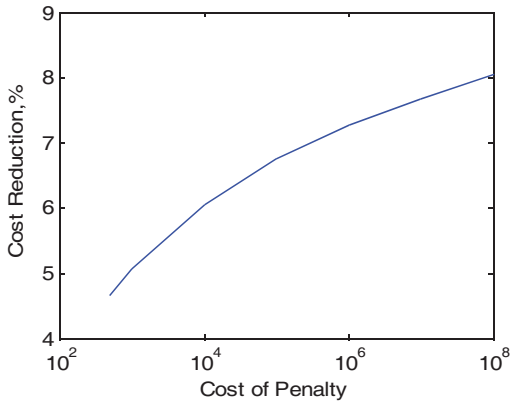


Figure 4: Effect of the weighting of the penalty on the cost reduction due to switching controller

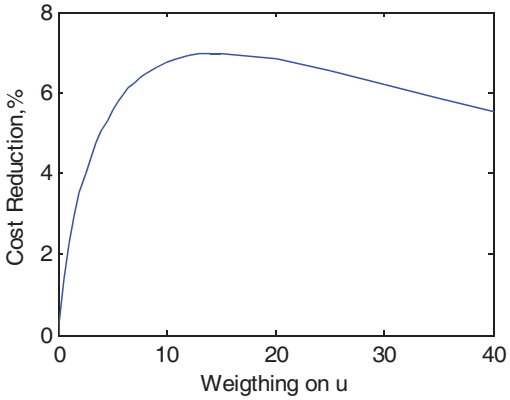


Figure 5 Effect of the input weighting on the cost reduction due to switching controller

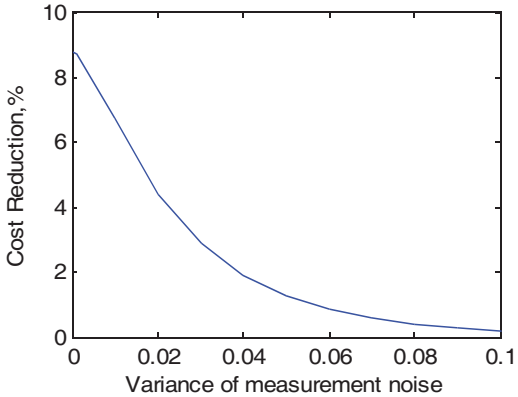


Figure 6: Effect of the measurement noise on the cost reduction due to switching controller

6. CONCLUSION

This paper showed the non-optimality of a symmetric PDF when the cost function was asymmetric and the manipulated variable was constrained. The result is derived using the analytical solution of the FPK equation for a scalar system and a switching controller. Finally, a numerical example was shown where the asymmetric PDF gave a better result than the symmetric one.

The importance of this result lies in the fact that it clearly demarks the cases where an asymmetric PDF is required. Also, a simple switching controller structure for PDF shaping is proposed that can be easily implemented in an industrial context. Finally, the analytical solution of the FPK equation is not only limited to PDF shaping, but could have more impact in the general context of stochastic optimization.

ACKNOWLEDGEMENTS

This work was supported by the Natural Sciences and Engineering Research Council of Canada (NSERC), Environmental Design Engineering Chair at École Polytechnique and Le Fonds Québécois de Recherche sur la Nature et les Technologies (FQRNT).

REFERENCES

Crespo, L.G., Sun, J.Q. (2002). Nonlinear Stochastic Control via Stationary Probability Density Functions. Proceedings of the American Control Conference. Volume 3. 2029-2034

Forbes, M.G. , Guay M., Forbes, J.F. (2004). Control Design for First-order Processes: Shaping the Probability Density of the Process State. *Journal of Process Control.* 14(4). 399-410

Guo, L., Wang, H. (2005*)*. Generalized discrete-time PI control of output PDF using square root B-spline expansion. *Automatica.* 41(1). 159-162

Jazwinski, A. H. 1970. *Stochastic Processes and Filtering Theory.* New-York: Dover Publications.376 p.

Kàrnỳ, M. (1996). Towards Fully Probabilistic Control Design. *Automatica.* 32(12). 1719-1722

Muske, K.(2003). Estimating the Economic Benefit from Improved Process Control. *Industrial Engineering Chemical Research.* 42(20). 4535-4544

Wang, H. (1998). Robust Control of the Output Probability Density Functions for Multivariable Stochastic Systems. *Proceedings of the 37st Conference on decision on control.* volume 2. 1305-1310

Wang, H. (2002). Minimum Entropy Control of Non-Gaussian Dynamic Stochastic Systems. *IEEE Transactions on Automatic Control.* 47(2). 398-403

Wang, H., Zhang, J. H. (2002). Control of the Output Stochastic Distributions via Lyapunov Function Analysis. *Proceedings of the 2002 IEEE International Conference on Control Applications.* 927-931.

Application of Near-infrared Spectroscopy in Batch Process Control

H. Lin*. O. Marjanovic. B. Lennox ***. A. Shamekh******

** Control Systems Centre, School of Electrical and Electronic Engineering, The University of Manchester, UK (e-mail: haishenglin318@yahoo.com.cn).*

*** Control Systems Centre, School of Electrical and Electronic Engineering, The University of Manchester, UK (e-mail: mchssom2@manchester.ac.uk).*

**** Control Systems Centre, School of Electrical and Electronic Engineering, The University of Manchester, UK (e-mail: barry.lennox@manchester.ac.uk).*

***** Electrical Engineering Department, University of Garyounis, Benghazi-Libya (e-mail: awshamekh@yahoo.com)*

Abstract: While batch processes are gaining ever increasing importance in the manufacturing industries, control of the product quality remains to be a serious challenge. To improve overall process understanding and control, new analytical techniques, such as Near-Infrared (NIR) Spectroscopy, are starting to be employed in industry. Currently, these techniques are primarily used for process monitoring purposes and have not yet been explicitly included in feedback control systems. This paper investigates the ability of three different control systems to adequately control a simulated batch reactor using the NIR spectra as feedback information such that the product meets quality specifications. The particular problem considered in this paper is adequate representation of the NIR spectrum using a single variable that is then controlled by employing Model Predictive Controller (MPC). It is shown that the resulting controller performances are highly variable if the controlled variable is chosen by selecting a single peak in the NIR spectrum to represent that variable. On the other hand, by using Principal Component Analysis (PCA) to extract information from all of the wavenumbers and represent it using a single composite variable, which is then controlled, it is shown that the process can be adequately regulated.

Keywords: Batch Process, Near Infrared Spectroscopy, Model Predictive Control, Chemical Reactor

1. INTRODUCTION

Batch processes are gaining ever increasing importance in the manufacturing industries. They are particularly prevalent in the polymer, pharmaceutical and specialty chemicals industries where the focus is on the production of low-volume, high-value added products. However, a major problem that is faced by those involved in batch processing is the application of reliable control systems. The characteristics associated with batch processes that make them particularly challenging to control include the presence of time-varying and nonlinear dynamics, multitude of unmeasured disturbances such as concentrations of various raw materials, and the presence of irreversible behaviour (Bonvin 1998).

This paper deals with the application of control systems to a chemical batch reactor for which the requirement to manufacture high quality product often translates into the control problem of tracking the reference temperature profile (Cott and Macchietto 1989). This is because the reaction rates involving raw materials, intermediates and products are highly dependent on the temperature. As a result, the composition of the product is also highly dependent on the reactor temperature. The reference profile design consists of characterising, in terms of the reactor temperature, the following three main stages of batch reactor operation: heating up the reactor; controlling the reactor temperature to meet the process requirement and then cooling down the reactor. However, temperature control of batch reactors can be a difficult task due to the process nonlinearities and the absence of the steady-state operation (Shinskey 1996). Aziz et al. (2000) analyzed the performance of different types of controllers in terms of their ability to track a reference profile of reactor temperature.

Even if the adequate temperature control system is in place and the reactor temperature does follow closely its reference profile, there is no guarantee that the final product will meet its specifications. For example, changes in the reaction rates and/or inclusion of a new raw material (as an impurity) can introduce new reaction pathways, which may cause the final composition of the product to change significantly. As a result, product quality can deteriorate even in the presence of a satisfactory temperature control system. Hence, it would be highly useful to construct a control system that would focus on regulating not the reactor temperature but some other variables that are much more directly related to product quality. As a result, such control system should be able to maintain high quality product in the presence of disturbances.

Near-infrared (NIR) spectroscopy represents a set of non-destructive analytical techniques that have been extensively used to extract chemical and physical information from a product sample based on scattered light (Reich 2005). NIR spectroscopy has been widely used in the pharmaceutical industry to test raw materials, control product quality and

monitor processes (M. Blanco 1998; Donald A. Burns 2001; Luypaert, Massart 2007). In the food industry there have been several applications of NIR spectroscopy being used for continuous process monitoring and control (Huang 2008).

Since the NIR spectra reflect the composition of the product, they represent excellent feedback information that could be used by control system to ensure the high quality of a product. So far NIR spectroscopy has been widely used for monitoring of manufacturing processes (Reich 2005; Jorgensen 2004; Scarff 2006). However, there is currently no publication proposing a method of explicitly using NIR spectra as feedback information to control the temperature of a reactor in order to ensure that the manufactured product conforms to high quality standards.

One clear problem in using NIR spectra as feedback information is the large number of variables that are needed to replicate information contained within the NIR spectrum. Arguably the number of variables should be equal to the number of spectral channels (wavenumbers) in order to completely characterise a given NIR spectrum. However, if this guideline is followed then the resulting control problem will potentially have several hundred controlled variables which could not be simultaneously controlled using typically only a handful or even just one or two manipulated variables.

In this paper, the problem of incorporating NIR spectrum as feedback information is addressed by using two different approaches. Both approaches utilise Model Predictive Control (MPC) framework but with a different definition of a controlled variable. The first approach is based on an idea of selecting wavenumber corresponding to one of the spectral peaks as a controlled variable. However, there are currently no clear guidelines regarding the selection of the peak to be considered as a controlled variable. The second approach is to use multivariate statistical analysis tools, namely Principal Component Analysis (PCA), in order to extract the information from NIR spectrum and represent it in a format of a single composite variable. This composite variable can then be regulated by means of a control system. Assessment of the controllers' performances is conducted using a simulated chemical batch reactor. The NIR spectrum is simulated by assuming that it is a linear combination of pure spectra related to individual compounds.

2. PRELIMINARIES

In this section the general concepts of Model Predictive Control (MPC) and Principal Component Analysis (PCA) are briefly introduced in order to facilitate the understanding of the control methodologies employed in the paper.

2.1 Model Predictive Control (MPC)

MPC (Maciejowski 2002) refers to a class of control algorithms that utilise an explicit process model to predict the future response of a plant. At each sampling instant, the MPC algorithm attempts to optimise future process behaviour by computing a sequence of adjustments that should be made to the manipulated variables. The first input in the optimal sequence is then implemented, and the entire calculation is repeated at the next sampling instant.

The key ingredient of the MPC controller is a prediction model used to forecast future process behaviour. In this paper the ARX structure (auto regressive with exogenous inputs) is chosen as the prediction model, and it is given as follows:

$$y(k) = -\sum_{i=1}^{n_y} a_i y(k-i) + \sum_{j=1}^{n_u} b_j u(k-j) + e(k) \qquad (1)$$

where $y(k)$ and $u(k)$ are the controlled and manipulated variable, respectively, at a sampling instant k. The model error is represented by $e(k)$. The order of the ARX model is determined by the values of n_y and n_u.

This cost function for the selection of the appropriate control action is given in (2).

$$J = \sum_{i=1}^{p} \alpha \left(y_r(k+i/k) - \hat{y}(k+i/k) \right)^2 + \sum_{j=1}^{m} \beta \Delta u \left(k+j-1/k \right)^2 \qquad (2)$$

Where J is the cost function to be minimized, p and m are the prediction and control horizons, respectively. y_r and $\hat{y}$ are the reference (set-point) values and estimated future output values, respectively, α and β are the weighting parameters for the controlled and manipulated variables, respectively. Finally, Δu is the change in manipulated variable (incremental control move) that is to be computed by the MPC algorithm.

The target of the cost function in (2) is to force the future output to track the reference trajectory over the specified prediction window p, while taking into account the balance between error energy and incremental control energy.

2.2 Principal Component Analysis (PCA)

The primary objective of Principal Component Analysis (PCA) is to capture the majority of variation present in data using a minimal number of composite variables, named principal components (PCs) (Johansson 2001; Berrar 2003). This dimensionality reduction is performed by exploiting the inter-dependence between measured process variables, such as individual wavenumbers in the NIR spectra.

For the analysis of spectroscopic data, such as that obtained from the NIR instruments, the power of PCA lies in its ability to condense the correlated information from hundreds of wavenumbers into a small number of mutually orthogonal principal components (PCs). Formally, PCA performs the following matrix decomposition:

$$\mathbf{X} = \mathbf{T}\mathbf{P}^{\mathbf{T}} + \mathbf{E} \qquad (3)$$

where $\mathbf{X}$ represents measured process data organised in n rows and m columns. PCA decomposes this data matrix into

the product of two matrices **T** and **P**, as shown in (3). **T** and **P** matrices contain as columns the so-called PCA scores and PCA loadings, respectively. **E** matrix represents the information contained within the matrix **X** that is not represented in the first nc principal components. Normally, each column of the data matrix **X** corresponds to a particular process variable, while the particular row is related to a specific sampling instant in time. In the context of NIR spectra, the columns of **X** represent specific spectral channels or wavenumbers while the rows contain data related to the whole NIR spectrum measured at a particular instance in time.

Due to the fact that the columns of the loadings matrix P are orthogonal, the expression for the calculation of scores is given as:

$$\mathbf{T} = \mathbf{XP} \tag{4}$$

It is the expression in equation (4) that will be utilised in this paper in order to condense information from hundreds of wavenumbers present in **X** into a single composite variable, namely the score associated with the first principal component.

3. CONTROL METHODOLOGY

3.1 Temperature Cascade Control (TCC)

A standard control problem in chemical reactor operation is that of controlling reactor temperature such that it follows a certain pre-computed reference trajectory, which should in turn ensure that the product quality will be satisfactory. Ultimately, reactor temperature is controlled by manipulating the flow of coolant or steam into the reactor's jacket. However, due to the presence of numerous disturbances, such as the feed temperature and the temperature of the incoming coolant, this control problem is addressed by employing two controllers in master-slave configuration, as shown in Fig.1.

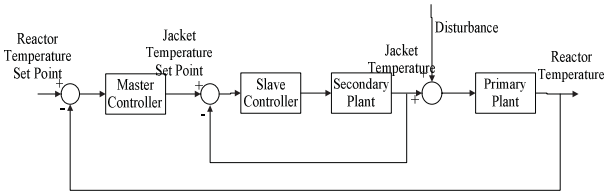


Fig. 1. Control of Reactor Temperature Using TCC System

The primary control loop, also known as the Master control loop, controls reactor temperature by adjusting the inlet jacket temperature set-point. The secondary control loop, known as the slave control loop, regulates jacket temperature by manipulating the flow of either coolant or steam into the jacket. Hence, the manipulating variable of the master control loop is the set-point for the slave control loop. This method of cascading controllers is very popular in the process industries and is particularly useful when there are disturbances associated with the slave controller's manipulated variable (Seborg 2004).

In this paper, a PI controller is used in the primary (slave) control loop while the PID controller is employed in the master (primary) control loop.

Note that the TCC system controls product quality implicitly, through the regulation of reactor temperature. The main problem with such implicit control arises with the occurrence of specific disturbances and process dynamics' changes, which adversely affect the underlying relationship between the reactor temperature and the product quality. As a result, optimal temperature profile will change. However, unless the optimal profile is calculated in real-time, TCC system will typically not have access to it. Instead, TCC will use existing reference trajectory, which is sub-optimal and may result in unsatisfactory product quality as demonstrated in the results section of this paper.

3.2 Wavenumber-Based MPC Control (Wn-MPC)

Spectroscopic instrumentation is being increasingly used to provide measurements, such as NIR spectra, that are in some way closely related to the product quality. By incorporating these measurements as feedback information into the control system the product quality control is addressed more explicitly when compared to the TCC scheme. One possible control system structure that incorporates NIR spectra as feedback information is shown in Fig. 2.

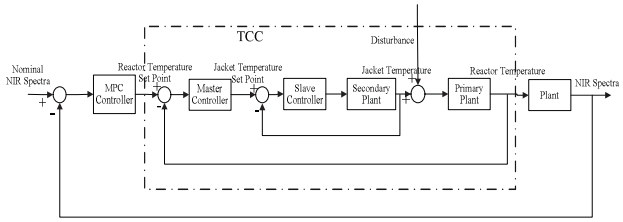


Fig. 2. Basic Structure of Wn-MPC Control

This new control structure incorporates the TCC system from Fig. 1 and augments it with the additional outer control loop, namely MPC control loop. The manipulated variable of the MPC controller is the reactor temperature set-point while controlled variables are the intensities of NIR spectra at a particular set of wavenumbers. Hence, within this control system structure, the TCC system can be viewed as a slave controller while MPC can be viewed as a master controller. This control system structure will be referred to as Wn-MPC.

The reference profile for the wavenumber is obtained by collecting NIR spectra from a 'nominal' batch, during whose progression no major disturbances were present and the standard TCC control scheme was used.

Since each wavenumber in the NIR spectra represents a candidate variable to be used as feedback information, there may be hundreds of potential controlled variables. Therefore, serious practical problem that arises when attempting to implement Wn-MPC is to decide on the set of wavenumbers that will be used as controlled variables. Currently, there are no clear guidelines as to which wavenumber should be selected for control purposes. In this paper a range of

wavenumbers was selected and their suitability was evaluated by incorporating them into Wn-MPC as controlled variables.

3.3 PCA Score-Based MPC Control (Sc-MPC)

In order to incorporate information from all of the wavenumbers into a feedback signal, a modified control system structure is used, as shown in Fig. 3.

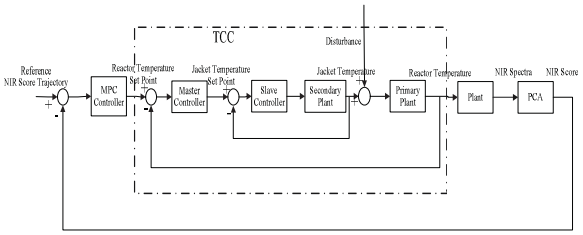


Fig. 3. Basic Structure of the Sc-MPC Control

This control scheme differs from Wn-MPC in that it includes a block containing the PCA model that pre-processes feedback information, namely NIR spectra. The result of PCA processing is a small set of variables, called scores, that contain information related to all of the measured wavenumbers. This is in contrast to Wn-MPC where the feedback information relates to only a few wavenumbers.

In this paper it is assumed that a PCA model is constructed using NIR spectra collected from a nominal batch. This nominal batch is run in the absence of any major disturbances using TCC control scheme. Hence, the resulting NIR spectra are assumed to represent reference profile that is to be replicated by Sc-MPC. In order to extract the main features from the highly multivariate NIR spectral data into a single variable, PCA model is applied. The resulting score trajectory is used as a reference profile that Sc-MPC is required to follow.

4. CASE STUDY

4.1 Chemical Reactor Simulation

This paper documents the application of three different control systems to a simulated chemical batch reactor taken from Cott (Cott and Macchietto 1989). The reactions taking place are given as follows:

$$A + B \xrightarrow{k_1} C;\ A + C \xrightarrow{k_2} D \qquad (5)$$

where *A*, *B* are the raw material, *C* is the desired product and *D* is the waste product, while k_1 and k_2 are the rates of the two reactions.

The control objective is to track the reactor temperature T_r reference trajectory by adjusting the jacket temperature T_{jsp}.

4.2 Disturbance Description

Three different control systems, described in section 3, were evaluated by injecting large disturbance and observing the control system response. Disturbance was chosen to be a reduction in a value of a reaction rate constant k_1 by 8%.

4.3 Prediction Model Identification

Training data for the Recursive Least Squares (RLS) algorithm was obtained using the TCC system structure, shown in Fig. 1. To excite the process dynamics, reference temperature trajectory was perturbed for three batches by adding a PRBS signal of amplitude 0.1 degrees C and switching time of 60 seconds.

In this particular case study ARX based prediction models were developed with $n_y = 2$ and $n_u = 80$. The data-driven identification method of RLS was used to develop dynamic models for both Wn-MPC and Sc-MPC controllers.

The output signal considered during the prediction model identification is the deviation of a controlled variable from its nominal trajectory. This controlled variable may be spectral intensity at the particular wavenumber (in the case of Wn-MPC control) or the value of the PCA score (in the case of Sc-MPC control).

4.4 Wavenumber Selection

In the case of Wn-MPC, candidate controlled variables were taken to be those wavenumbers that corresponded to a local peak of the measured NIR spectrum. In this particular case study the wavenumbers corresponding to the local peaks in the NIR spectra and, therefore, representing the candidate controlled variables were 2, 77, 98, 127, 161 and 232, as illustrated in Fig. 4.

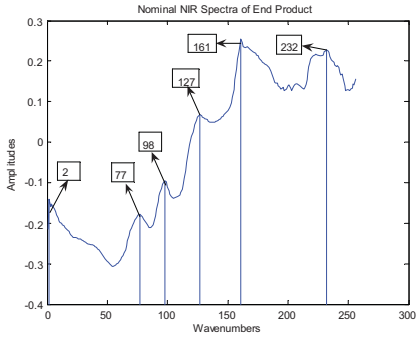


Fig. 4. Selection of spectral peaks as controlled variables

For each of these wavenumbers prediction model was identified and the corresponding MPC controller was constructed and evaluated. The corresponding controllers are designated with a chosen wavenumber written within brackets following a label Wn-MPC. For example, Wn-

MPC(127) designates Wn-MPC controller that utilises wavenumber 127 as the controlled variable.

4.5 PCA Model Development

A PCA model was developed using NIR spectra collected from a single nominal batch. The first PCA score captured 93.8% of the variation present in the NIR spectra and was used as a reference trajectory in the subsequent implementation of Sc-MPC controller. The loadings vector associated with the first PCA score was then used in real-time to compute score value from the measured NIR spectra according to equation (4).

4.6 Results and Discussion

For each controller (TCC, Wn-MPC and Sc-MPC) the process was perturbed using the identical large disturbance described in section 4.2. The resulting NIR spectra that corresponded to particular controllers along with the reference spectrum are plotted in Figures 5 and 6.

Fig. 5 shows the NIR spectra obtained when the controllers used to regulate the batch reactor were TCC, Sc-MPC and Wn-MPC(77). Sc-MPC can be seen to outperform both TCC and Wn-MPC(77). In fact, the NIR spectrum obtained when using Sc-MPC controller was found to be very similar to the reference spectrum, as shown in Fig. 5. On the other hand, both TCC and Wn-MPC(77) clearly failed to reject the disturbance as evidenced by considerable deviation of their respective NIR spectra from the reference spectrum.

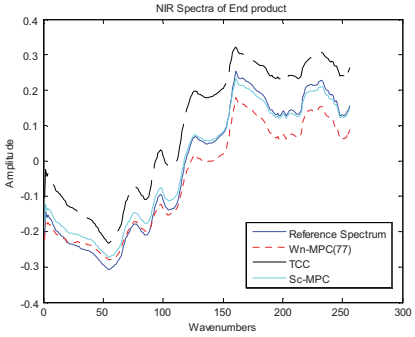


Fig. 5. NIR spectra of end product obtained when using TCC, Wn-MPC(77) and Sc-MPC

The reason for the discrepancy in performance between the TCC and Sc-MPC lies in the fact that the TCC control system does not consider NIR spectra as its feedback information and, furthermore, its reference temperature profile is not adjusted to account for the presence of the large disturbance, which has modified the underlying relationship between temperature and product quality. On the other hand, Sc-MPC explicitly considers regulation of the NIR spectra by using the composite of spectral measurements as its feedback information. Wn-MPC(77) also delivered sub-optimal performance because the spectral data contained in wavenumber 77 appeared not to be sufficient to characterise the majority of information contained in the entire NIR spectrum. Wn-MPC(77) is an example of Wn-MPC controller with its controlled variable obtained by randomly selecting one of the prominent peaks in the NIR spectrum, which is not an unlikely scenario in real applications.

The performances obtained by controlling NIR trajectories at different wavenumbers (77 127 161) using Wn-MPC controllers change largely, as demonstrated in Fig. 6.

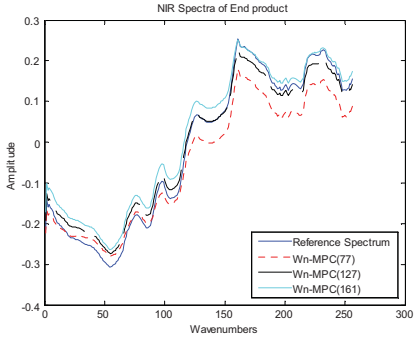


Fig. 6. NIR spectra of end product obtained when using Wn-MPC(127), Wn-MPC(161) and Wn-MPC(77)

The sum of square errors of the NIR spectra and its nominal values by Wn-MPC at every wavenumber are calculated and showed in Fig. 7. This figure shows a large variation in performance achieved by Wn-MPC controllers that utilise different wavenumbers (2, 77, 98, 127, 161, 232) as their controlled variables.

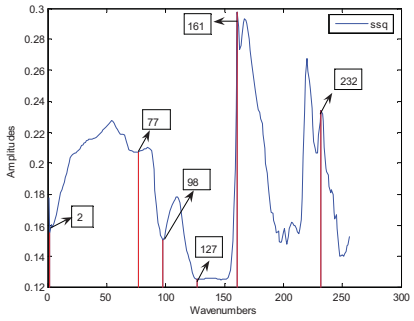


Fig. 7. The sum of square errors (ssq) by Wn-MPC at different wavenumbers

Even if the Wn-MPC is used with an optimally selected wavenumber, which is wavenumber 127 in this particular case study, the resulting control performance was found to be very similar to the performance of the Sc-MPC controller. This is demonstrated in Fig. 8 where the NIR spectra shown were obtained when the process was being controlled using Sc-MPC and Wn-MPC(127).

Hence, the improvement in performance delivered by Wn-MPC(127) is not considerable while the trial-and-error procedure involved in selection of the wavenumber to be controlled may be prohibitively time-consuming and expensive. On the other hand, Sc-MPC delivered satisfactory

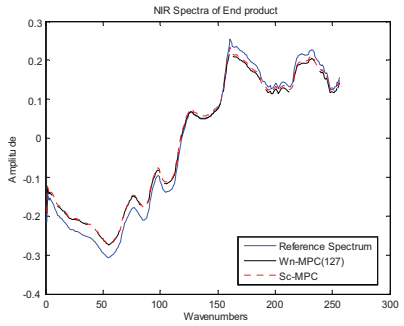


Fig. 8. NIR spectra of end product obtained when using Wn-MPC(127) and Sc-MPC

performance that was similar to that of the Wn-MPC(127) controller. In addition, the controlled variable is automatically selected requiring no trial and error in the case of Sc-MPC. Hence, Sc-MPC controller was found to require minimal user interaction when selecting appropriate controlled variable while also delivering a highly satisfactory performance.

Observed variability in performance can be explained by the fact that all of the considered Wn-MPC controllers focus on the feedback information contained within a single wavenumber. Hence, there may be cases where a chosen wavenumber conveys little information related to other segments of the overall NIR spectrum, such as the wavenumber 77. In these cases the resulting Wn-MPC will not deliver satisfactory performance, as is the case with Wn-MPC(77). Similarly, there may be cases where a single wavenumber does reflect many of the features of the entire NIR spectrum, such as the wavenumber 127. Resulting controller, namely Wn-MPC(127), will then deliver a satisfactory performance.

5. CONCLUSIONS

This paper investigated the ability of three different control systems to adequately control a simulated batch reactor using the NIR spectra as feedback information such that the product meets quality specifications. The first of the three controllers ignored the presence of NIR spectra and was solely concerned with the regulation of reactor temperature such that it follows pre-specified reference trajectory. This controller was found to be inadequate when the large disturbances altered the underlying relationship between reactor temperature and product quality. The other two controllers utilised aspects of the measured NIR spectrum in their formulations. One of these two controllers used spectral intensities at specific wavenumbers (spectral channels) that corresponded to local peaks in NIR spectra as feedback information and was referred to as Wn-MPC. The other controller used multivariate statistical tool, namely Principal Component Analysis (PCA) in order to extract the main features present in all of the wavenumbers and condense this information into a single composite variable that was controlled. This controller was referred to as Sc-MPC. Results of implementing these three controllers on a simulated batch reactor reveal that the Sc-MPC achieved satisfactory control while also requiring no user interaction when deciding on the variable to be controlled. On the other hand, performance achieved by Wn-MPC was found to be highly dependent on the choice of the wavenumber that is to be controlled. However, due to the lack of rigorous guidelines when selecting appropriate wavenumber and the resulting trial and error necessary to determine optimal wavenumber, it is questionable whether Wn-MPC can be used as a practical solution in industrial process control area.

REFERENCES

Aziz, N., Hussain, M.A., and Mujtaba, I.M. (2000). Performance of different types of controllers in tracking optimal temperature profiles in batch reactors. Computers & Chemical Engineering, 24(2-7), 1069-1075.

Berrar, D.P., Dubitzky, W., and Granzow, M. (2003). Singular value decomposition and principal component analysis. A Practical Approach to Microarray Data Analysis, Kluwer: Norwell, MA.

Blanco, M., Coello, J., Iturriaga, H., Maspoch, S., and Pezuela, C.d.l. (1998). Near-infrared spectroscopy in the pharmaceutical industry. Analyst, 123, 135-150.

Burns, D.A., and Ciurczak, E.W. (2001). Handbook of Near-Infrared Analysis, CRC.

Cott, B.J., and Macchietto, S. (1989). Temperature Control of Exothermic Batch Reactors Using Generic Model Control. Industrial & Engineering Chemistry Research, 28(8), 1177-1184.

Huang, H.B., Yu, H.Y., Xu, H.R., and Ying, Y.B. (2008). Near infrared spectroscopy for on/in-line monitoring of quality in foods and beverages: A review. Journal of Food Engineering, 87(3), 303-313.

Johansson, E., Ettaneh-Wold, N., and Wold, S. (2001). Multi- and Megavariate Data Analysis. Umetrics Academy.

Jorgensen, P., Pedersen, J.G., Jensen, E.P., and Esbensen, K.H. (2004). On-line batch fermentation process monitoring (NIR): introducing 'biological process time'. Journal of Chemometrics, 18(2), 81-91.

Luypaert, J., Massart, D.L. (2007). Near-infrared spectroscopy applications in pharmaceutical analysis. Talanta, 72(3), 865-883.

Maciejowski, J. (2002). Predictive Control With Constraints, Addison Wesley Longman.

Reich, G. (2005). Near-infrared spectroscopy and imaging: Basic principles and pharmaceutical applications. Advanced Drug Delivery Reviews, 57(8), 1109-1143.

Scarff, M., Arnold, S.A., Harvey, L.M., and McNeil, B. (2006). Near Infrared Spectroscopy for bioprocess monitoring and control: Current status and future trends. Critical Reviews in Biotechnology, 26(1), 17-39.

Seborg, D.E., Edgar, T.F., and Mellichamp, D.A. (2004). Process dynamics and control. Wiley.

Shinskey, F.G. (1996). Process control systems. McGraw-Hill, New York, USA.

Profitability and Re-usability: An Example of a Modular Model for Online Optimization

Margret Bauer*. Moncef Chioua.*
Jörg Schilling*.
Guido Sand*. Iiro Harjunkoski*.

**ABB Corporate Research, Ladenburg 68526, Germany*
(Tel: +49-6203-716284; e-mail: margret.bauer@de.abb.com)

Abstract: In this article, we describe the development of a modular online optimization solution. This solution can be configured to deal with different plant layouts and therefore allows for a re-usable and hence profitable advanced optimization solution. The industrial process for which the model was developed is divided into separate stages. Each stage represents a production step, which may or may not be present in a particular plant. Inputs and outputs of each stage are defined in a flexible way to ensure that the sequence of the production stages can vary and can be easily connected. Optimization results are shown for two alternative plant configurations and are discussed together with the benefits and cost that come with the pursuit of a modular solution.

Keywords: Online optimization, modularity, advanced solutions, nonlinear optimization, multi-stage processes, configuration.

1. INTRODUCTION

Most advanced industrial control and optimization solutions are developed on a case-by-case basis and are tailored to a production process at a particular plant. A large part of the implementation is spent on expert manpower; to model the plant behaviour with a deterministic or stochastic model and to adjust the control algorithm of the model (Bauer and Craig, 2008).

From a vendor's point of view, the business case of developing advanced solutions will often only achieve a breakeven if the model and the control algorithm can be re-used and installed multiple times without long and tedious adaptation of the model and algorithm. The development cost can then be split between the several implementation projects and the solutions is offered at a price that will result in an acceptable net present value (NPV) – both for the vendor and the customer.

Achieving re-usable and thus profitable advanced control and optimization solutions, however, are as rare as hen's teeth. Darby and White (1988) point out that this can be a question of modelling the process with one single model or in a decentralized approach. A modular approach is often easier to implement and maintain, especially when model updates are required.

In some industrial plants, production stages resemble each other and show enough common features to be generalized by a basic building block. A modular approach is achieved by configuring and connecting the basic building blocks. Other processes might be constructed by the same basic building block with different parameterization that is configured to adjust to the plant layout, including the use of a different number of blocks.

In this article, we describe a modular nonlinear real-time optimization approach that maximizes the throughput in a production process. Nonlinear online optimization is still not applied as widely as linear optimization problems in industry but some applications have been reported (De Gouvea and Odloak, 1998, Jockenhövel et al., 2003). The approach has been developed for an industrial environment to be implemented for online application.

Here, we describe an example of a production process for which a modular model will be developed. This model is derived using the following steps.

- Identify a basic building block;
- Define the block equations;
- Define the block connections;
- Define the process' objective function;
- Package the basic building block.

If a model is available in such a way, it significantly reduces the effort required to adapt the model to a new plant configuration. The solution can be re-used once implemented on an appropriate platform. One of the keys to a profitable advanced solution is furthermore the easy configuration and implementation of any modelling approach. An example is ABB's Expert Optimizer that provides the framework for implementing hybrid model predictive controllers. An example of such an implementation is given by Stadler and co-workers (2007).

The paper is structured as follows. In the next section, the process is described and two examples of possible configur-

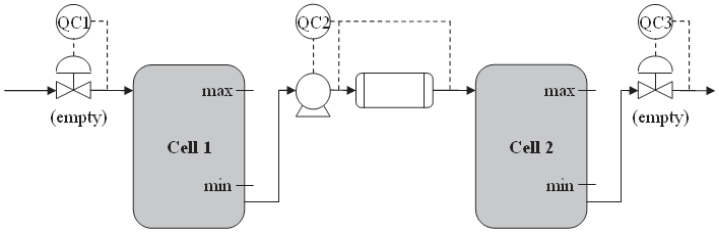


Fig. 1. Process schematic of Configuration I consisting of two cells.

ations are given together with the objective of the optimization problem. The different types to which the basic building block can be configured are identified and listed. Section 3 describes the model derivation along the steps stated above. Results are given in Section 4 together with a discussion on the advantages and disadvantages of the modular solution.

2. PROCESS DESCRIPTION

The continuous process under consideration consists of a set of sequential cells. An example of a configuration of this process is shown in Fig. 1. The medium flows from one cell with a certain flow rate. Valves or pumps control the flow rate into each cell. The flow rate is limited and so is the volume contained in the cell. What makes the process particular and difficult to control is the fact that the medium to be processed changes almost completely and abruptly. The medium consists, in fact, of different separate products with different attribute affecting the flow rate. The products do not mix but instead are processed successively. Thus, the flow rate changes whenever the next product enters a cell. A sensor therefore measures the consistency at the entry of the cell and the flow rate is adjusted accordingly.

The particular are characteristics concerning the operation of the two cells. The first cell has to be emptied before a new product enters it. Thus, the valve or alternatively pump is closed off for a period of time until the tank has been emptied to a certain level. The second cell is preceded by a heat exchanger that warms up the medium before it enters the cell. While the first part of the medium is in the heat exchanger, the heat has to be adjusted to a certain temperature and the flow rate has to be lower than its normal maximum. In any configuration, an outflow valve controls the last flow in the process. The outflow is interrupted as one product is filled into a container, the container is sealed off, removed and a new container is placed in this position.

Altogether, there are four basic cell types, each with or without a heat exchanger and/or emptying during transition, as listed in Table 1. A plant can consist of a number of sequential cells, ranging from two up to about eight cells.

The process is very difficult to control as all cells interact and disturbances directly travel through the process. Minimum and maximum constraints of the level in the cells and of the flow rate are hard and cannot be violated without causing a complete shut-down of the plant. An important process characteristic is that the flow rates have to be constant while one product is filled into a cell. The level is therefore constantly increasing or decreasing, depending on the difference between the in- and outflow of the cell. If in- and outflow are identical for a certain period of time the level stays constant for that period of time. Determining the optimal set-points is therefore crucial for an uninterrupted operation of the process that also maximizes the throughput.

2.1 Configuration 1

The process can have different setups of the basic cell types described in Table 1. The one such configuration is shown in Fig. 1 and, as described earlier, consists of two cells where Cell 1 has to be emptied before the next product can enter it (Case B). During this period, the valve is closed off. Cell 2 is preceded by a heat exchanger (Case C). The consistency is measured before and after the heat exchanger so that a product change is noticed thereafter so that the flow rate can be adjusted when the medium enters and exits the heat exchanger. The opening and closing of the cells in- and outflow depends on the different material and the control is indeed already very complex when considering only these two cells.

2.2 Configuration II

Fig. 2 shows the process schematic of the second configuration to be investigated in this paper. Here, three cells are connected where the first cell has to be emptied after a product changeover (Case B). The second cell adjusts the speed according to a product change but neither has it a heat exchanger nor is it emptied (Case A). The third case has a heat exchanger but does not have to be emptied between product changeovers (Case C). The plant parameters such as maximum flow rate and cell volume differ from Configuration I. This naturally affects the different operational routines.

Table 1. Alternative cell types

		Emptying during transition	
		No	Yes
Heat exchanger	No	Case A	Case B
	Yes	Case C	Case D

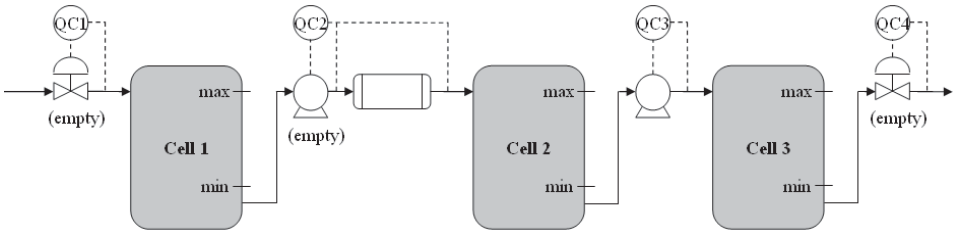


Fig. 2. Process schematic of Configuration II consisting of three cells.

2.3 Online process optimization

The described process is somewhat different to a standard continuous chemical process. Here, we deal with different products in operation with changing attributes. The controller has to deal with different states that alter with a product changeover: emptying, heating and normal operation. The aim of this study is to determine the flow rate set-points for a sequence of products under the given constraints, that is, limits of the constant flow rates and minimum and maximum cell levels. The objective is to maximize the outflow of the last cell. New setpoints are determined repeatedly, either:

a) Time based, that is, on a fixed time grid for example every ten seconds;
b) Event based, that is, if a defined event occurs, for example if a new product enters the process.

If an event occurs, the time grid is reset and restarted after the event. The high update frequency requires a fast result from the online optimization routine.

3. MODULAR MODELLING

The processes described in the previous section can be modelled as one single problem since the number of variables is limited. However, in order to re-use the optimization solution for both Configuration I and II and possibly other configurations, it is advantageous to model a basic building block and then configure and connect the blocks using the same description and connections. In the following, the basic building block is identified, the equations are identified, connections established and the objective function for the complete process is derived.

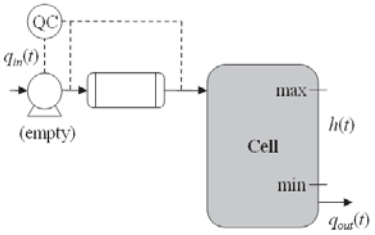


Fig. 3. Basic building block from which Configuration I and II can be constructed.

3.1 Identification of basic building block

By looking at Fig. 1 and Fig. 2 one can easily identify the repeating elements in the process such as the cell and inflow control. A basic building block that describes all cells and their inflow control is shown in Fig. 3. It consists of one cell and the adjustable flow rate of the inflow. The heat exchanger preceding the cell is included in the generic block and its parameters and equations will be set to zero if no heat exchanger is present. Variables are also introduced for emptying the cell. In case that the cell does not have to be emptied, these variables are also set to zero.

3.2 Building block equations

First, the decision variables to be optimized are introduced for each cell. These variables are noted with lower case letters and include the following. There are p products with $p=\{1...P\}$.

$h(t)$	Cell level
$q_p^{in}(t)$	Inflow to cell
$q_p^{out}(t)$	Outflow of cell
τ_p	Time duration during which the product flows into the cell
τ_p^{empty}	Time duration during which the cell and the heat exchanger are emptied
τ_p^{warmup}	Time duration during which the product is warmed up in the heat exchanger

The flow rates are fixed for the duration while processing product p. If the flow rate is constant then the level is a linear function. The durations are auxiliary variables.

Parameters to be configured for each cell are as follows.

V_p	Volume of product p
V^C	Volume of cell C
V^{HE}	Volume of the heat exchanger
$Q_p^{\min}$, $Q_p^{\max}$	Minimum and maximum flow rate

$H^{\min}$, $H^{\max}$ Minimum and maximum cell level

If no heat exchanger is located ahead of the cell (Case A and B) then volume V^{HE} is set to zero.

The duration during which the product flows into the cell is defined as the volume of product *p* minus the volume of the heat exchanger divided by the flow rate of product *p*. When the next product enters the cell the flow rate changes to q_{p+1}^{in}.

$$\tau_p = \frac{V_p - V^{HE}}{q_p^{in}} \tag{1}$$

This equation is nonlinear and thus can cause difficulties for most solvers. It is therefore necessary to reformulate this equation as well as the following into a bilinear form by multiplication ($\tau \cdot q_p^{in} = V_p - V^{HE}$).

When emptying the cell, the valve or pump is closed off and the inflow rate is hence set to zero. The duration during which the tank is emptied is determined by the outflow rate. The volume to be emptied is the volume in the cell plus the volume in the heat exchanger.

$$\tau_p^{empty} = \frac{V^C + V^{HE}}{q^{out}} \tag{2}$$

The duration during which the product is warmed up is defined by the volume of the heat exchanger divided by the inflow rate. It is independent from the outflow rate as one might initially expect.

$$\tau_p^{warmup} = \frac{V^{HE}}{q_p^{in}} \tag{3}$$

The level is proportional to the difference between the in- and outflow rate. The proportional coefficient depends on the area of the cell.

$$h(t) \sim q^{in}(t) - q^{out}(t) \tag{4}$$

Eq. (1)–(4) describe the dynamics of the cell. Inflow, outflow and level have to be defined for each time point for which a switch occurs, that is, when a new product reaches a measuring point. There are two measuring points in case of the presence of a heat exchanger, one before and one after. At these switching points, the level reaches its minimum or maximum value as the function increases and decreases only linearly. If the cell level does not violate the constraints at two consecutive switching points, it will not violate the constraints at any time between those switching points. The reformulation of the inflow, outflow and level for these switching points is rather cumbersome in notation but straight forward otherwise. It will therefore not be detailed in this article.

In addition to the equations describing the process dynamics there are also constraints that determine the operation of the cells. These constraints are considered for the inflow rate q^{in} and for the cell level *h*.

$$Q_p^{\min} \le q_p^{in}(t) \le Q_p^{\max} \tag{5}$$

Table 2. Parameter adaptation for cell types

		Emptying during transition	
		No	Yes
Heat exchanger	No	$V^{HE} = 0 ; \tau_p^{empty} = 0$	$V^{HE} = 0$
	Yes	$\tau_p^{empty} = 0$	none

$$H^{\min} \le h(t) \le H^{\max} \tag{6}$$

The constraints on the outflow rate do not have to be considered as they are defined in the successive cell.

3.3 Connection of building blocks

The cells are connected by the flow through the process. The connection is formulated by equating the outflow rate of cell *C* with the inflow rate of the subsequent cell *C*+1.

$$q^{out,C}(t) = q^{in,C+1}(t) \tag{7}$$

3.4 Objective function

The objective of the optimization problem is to maximize the throughput of the process. As the throughput is determined by the outflow rate of the last cell, this is the quantity to be maximized.

$$\max \int q^{out,C=C_{last}}(t)dt \tag{8}$$

Alternatively, it is possible to minimize the sum of all durations defined in Eq. (1)–(3). In some cases, a better solution is obtained by maximizing the flow rate in all cells and not only the one of the last. This is particularly valid for very short product sequences as the first cells may not process with the highest rate as the finishing of the product sequence in the last cell does not depend on it. As a result, the objective function is set to Eq. (8) plus an additional term including the flow rates in the other cells multiplied by a weighting factor smaller than one.

3.5 Configuration

The basic building block can be packaged into a stand alone function with input and output variables as shown in Fig. 4.

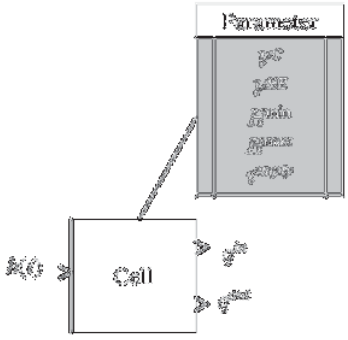


Fig. 4. Configuration block for implementation.

The parameters have to be set for each block. The differentiation between the cell types as given in Table 1. leads to a parameter configuration that is summarized in Table 2. The final step is the connection between the in- and outflows of the consecutive cells as given in Eq. (7) to derive the complete plant setups of Fig. 1 and Fig. 2.

4. OPTIMIZATION RESULTS

The same model is applied for both Configuration I and II with only changes to the parameters and the number of cells. The optimization problem was implemented in GAMS and as a nonlinear program solved with CONOPT. CONOPT is based on the generalized reduced gradient method which transforms inequality constraints into equality constraints by introducing slack variables. The solver then searches along the steepest slope of the super-basic variables.

In some instances, nonlinear models can easily lead to infeasibility. However, as the initialization of the optimization routine is already close to the optimum results can be found reliably. Upper constraints of flow rate and level are used as initial values. The solution is also not necessarily the global optimum. This decision is left to the writer of the GAMS code and the model developer.

Fig. 5 shows the results of Configuration I of Fig. 1. Here, two cells were connected with a heat exchanger between the cells. The first cell had to be emptied/flushed in preparation for a product change. The outflow of the process was also interrupted to allow the product to be filled into tanks. The tanks have to be removed and the process outflow closed off during that period. The process operation can be best seen in the flow rates. The left hand side of Fig. 5 shows the three flow rates, q_1-q_3 for three products. The first flow rate is the process inflow and is interrupted each time a new product enters the cell. The maximum flow rate into the first cell is large for both products, however, the set point is set to a lower level to not exceed the level in the cell as the outflow of the first cell is limited by significantly lower constraints. Flow rate $q_2(t)$ is at its maximum constraint as it poses, together with $q_3(t)$ the bottle neck in the process. While the first part of a product is processed in the second cell, a reduced flow rate is applied. The flow rate $q_3(t)$ is at a higher value than $q_2(t)$ but a stop time interrupts the flow during the product changeover.

The cell levels shown in the right hand side of Fig. 2 indicate that the cell level of the first cell is in the region of its upper limit, i.e., the cell is filled during most time of the operation. The second cell, on the other hand, hits on some occasions the lower constraints.

The results of the optimization routine of Configuration II are shown in Fig. 6. Here, four flows and three cell levels are shown for three products. The first cell is emptied with every product changeover, as can be seen in $q_1(t)$. A heat exchanger is placed ahead of the second cell which affects the flow rate $q_2(t)$. The flow rate $q_3(t)$ changes only with the different products while the last flow rate includes stops during which a new tank is replaced. The stop times for the tanks are constant.

The last cell level is somehow cyclic as the cell is emptied for the new product and then filled up again. The cycle would be repeated if the optimization would have been carried out for more products. The level $h_1(t)$ shows similar features while $h_2(t)$ decreases and then stays at its minimum level for the last product as the in- and outflow rates are identical.

Because of the modular approach, the same model could be re-used for both configurations. The model changes are only changes to the input parameters but not the model equations. In both cases, the outflow rate was optimized which ensures that the flow rates are at their maximum for the bottleneck cells.

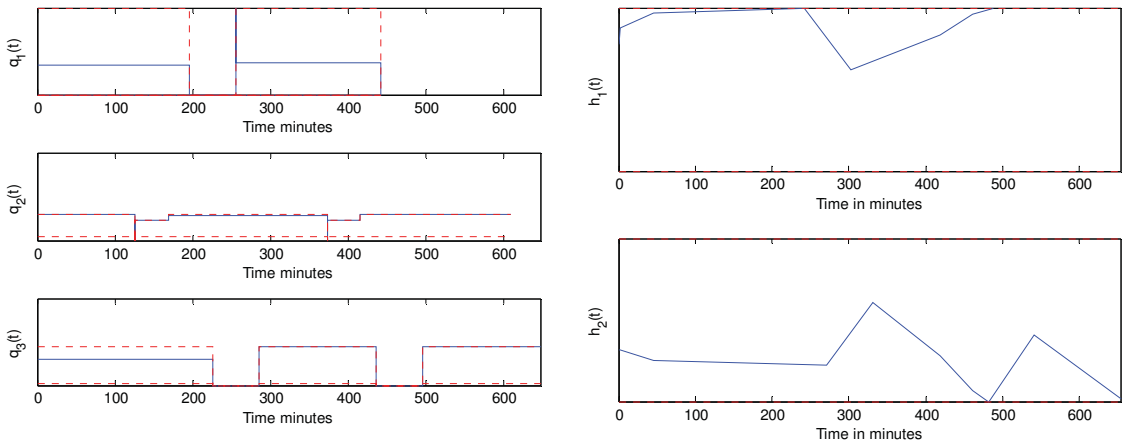


Fig. 5. Results for process Configuration I: cell inflow rates $q(t)$ and levels $h(t)$. Dashed lines indicate the upper and lower constraints.

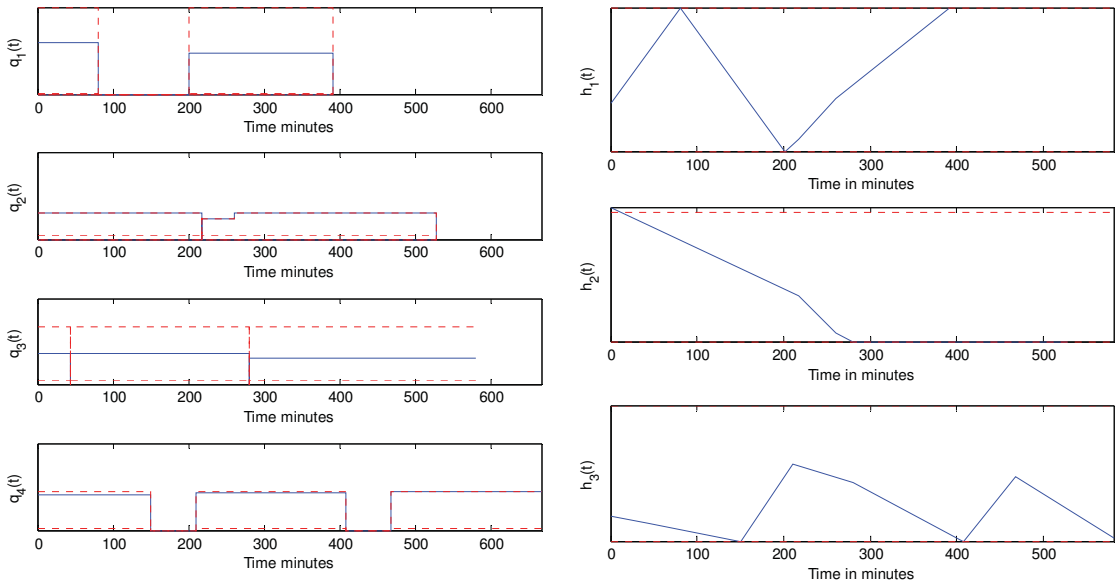


Fig. 6. Results for process Configuration II: cell inflow rates $q(t)$ and levels $h(t)$. Dashed lines indicate the upper and lower constraints.

5. CONCLUSIONS

To achieve a viable business model of advanced control and optimization solutions the modelling, implementation and maintenance effort has to be as small as possible. Re-usable solutions do not only decrease the modelling effort but also make it easier to maintain the solution as the development and commissioning engineers have to be familiar with one solution type. It is therefore attractive to build a modular solution that can be applied to different process setups. In this article a modular solution for a process with different configurations has been derived. The model has been applied to industrial processes and is currently in the process of deployment.

The key steps followed were as follows. A basic building block was identified and the equations introduced, including the connection between the blocks. Optimization results were discussed. Deriving this kind of modular approach is key when developing solutions that can be easily adapted and therefore have the potential to become a business success.

REFERENCES

Bauer, M. and Craig, C., 2008. Economic assessment of advanced process control – A survey and framework. *Journal of Process Control*, 18, 2-18.

Darby, M.L. and White, D.C., 1988. On-line optimization of complex process units. *Chemical Engineering Progress*, 84, 51-59.

Jockenhövel, T., Biegler, L.T. and Wächter, A., 2003. Dynamic optimization of the Tennessee Eastman process using OptControlCentre. *Computers & Chemical Engineering*, 27, 1513-1531.

De Gouvea, M.T. and Odloak, D., 1998. One-layer real time optimization of LPG production in the FCC unit: Procedures, advantages and disadvantages. *Computers & Chemical Engineering*, 22, S191-S198.

Stadler, K.S., Wolf, B. and Gallestey, E., 2003. Model predictive control of the calciner at Holcim's Lägerdorf plant with the ABB Expert Optimizer. *ZKG International*, 60, 66-67.

A PID automatic tuning method for distributed-lag processes ⋆

Massimiliano Veronesi * **Antonio Visioli** *

* *Dipartimento di Elettronica per l'Automazione, University of Brescia, Italy*
e-mail: antonio.visioli@ing.unibs.it

Abstract: In this paper we present an automatic tuning methodology for PID controllers for distributed-lag processes. The technique is based on the evaluation of a closed-loop set-point or load disturbance step response and it can be therefore employed with process routine operating data. Further, a performance assessment index is also proposed in order to establish when the performance of a PID controller can be improved by retuning it according to the proposed method. Simulation results show the effectiveness of the approach.

1. INTRODUCTION

Distributed-lag processes are frequently encountered in the process industry. For example, transmission lines, heat exchangers, stirred tanks and distillation columns might have a dynamic characteristic so that they can be modelled as an infinite series of infinitesimally small interacting lags and therefore as a distributed lag (Shinskey, 1994). Despite this fact, this kind of processes are rarely considered in the academic literature (Shinskey, 2002), with the notable exception of the works of Shinskey (see, for example, (Shinskey, 2001)). Therein a tuning tule for Proportional-Integral-Derivative (PID) controllers has been proposed based on process parameters that are obtained by evaluating an open-loop step response.
Indeed, many tuning rules have been developed for PID controllers (O'Dwyer, 2006) and the great majority of them are based on a first-order-plus-dead-time (FOPDT) or second-order-plus-dead-time (SOPDT) model of the process that can be obtained typically by evaluating an open-loop step response. However, this experiment can be time-consuming and, above all, it can imply that the normal process operations are stopped, which is obviously not desirable. For this reason, automatic tuning methodologies have been developed also based on closed-loop experiments, usually by considering a relay-feedback experiment (Yu, 1999).

In this paper we present a methodology for the automatic tuning of PID controllers for distributed-lag processes which is based on the evaluation of a *closed-loop* set-point or load disturbance step response. In particular, we assume that a (possibly badly tuned) PID controller is operating and the evaluation of the step response is employed to retune the PID controller if the achieved performance is not satisfactory, as in (Veronesi and Visioli, 2009). In order to assess the performance of the controller, a performance index is proposed, so that the methodology can be applied both for tuning-on-demand (namely, the controller is tuned after an explicit request of the operator) and for self-tuning (namely, the controller itself determines that the control performance is not satisfactory and a new tuning is provided). It is worth stressing that the tuning rule applied is devoted to the load disturbance rejection task which is usually of main concern in the above mentioned processes.

The paper is organised as follows. A model for distributed-lag processes is given in Section 2. The autotuning method is presented in Section 3, where we explain how the relevant process parameters can be obtained and how the PID parameters can be selected. Finally, the practical implementation of the method is addressed. Simulation results are given in Section 4, and conclusions are drawn in Section 5.

2. MODELLING

A distributed-lag process can be described by the following transfer function (Shinskey, 1994)

$$P(s) = \frac{2\mu}{e^{\tau s} + e^{-\tau s}} = \frac{\mu}{\cosh\sqrt{\tau s}} \tag{1}$$

where μ is the process gain. The hyperbolic cosine can be expanded into an infinite-product series, so that we obtain

$$P(s) = \frac{\mu}{[1 + (2/\pi)^2\tau s][1 + (2/3\pi)^2\tau s][1 + (2/5\pi)^2\tau s]\cdots} \tag{2}$$

It is worth noting that the sum of all time constants, denoted as T_0, is equal to 0.5τ. If a unit step is applied to the process input, the sum of all time constants can be estimated easily as the time the process variable takes to attain the 63.2% of its steady-state value (see Figure 1). Then, the process gain can be estimated easily by considering the steady-state value of the process output and the amplitude of the step input (Visioli, 2006a). However, the open-loop experiment can be time-consuming and, in order to perform it, it can be necessary to stop the routine process operations. Thus, we propose a method to estimate the value of T_0 and of the process gain μ with a closed-loop experiment, namely by employing a PID controller with any values of the parameters (provided that the closed-

⋆ This work was partially supported by MIUR scientific research funds.

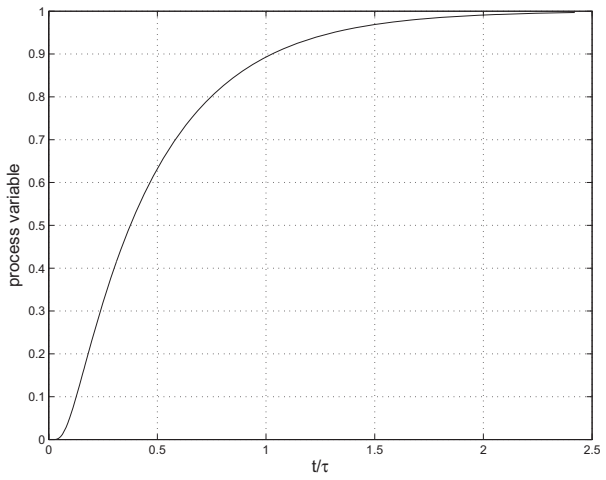


Fig. 1. Open-loop step response of a distributed-lag process. The process variable attains the 63.2% of its steady-state value at time $t = T_0 := 0.5\tau$.

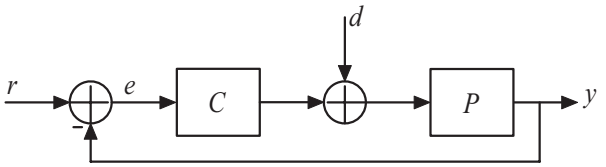


Fig. 2. The considered control scheme.

loop system is asimptotically stable).
Note that, for the purpose of simulation, transfer function (2) can be written as

$$P(s) = \frac{\mu}{\prod_{i=0}^{n-1}\left[1+\left(\frac{2}{(2i+1)\pi}\right)^2 \tau s\right]} \tag{3}$$

with n at least equal to 20, because the dynamics of the process does not change significantly for $n > 20$ (Shinskey, 2001).

3. AUTOMATIC TUNING

3.1 Estimation of the process parameters from set-point step response

We consider the unity-feedback control system of Figure 2 where the process P is controlled by a PID controller whose transfer function is in series ("interacting") form:

$$C(s) = K_p\left(\frac{T_i s + 1}{T_i s}\right)(T_d s + 1). \tag{4}$$

The series form has been chosen for the sake of simplicity, however, the use of other forms is straightforward by suitably applying translation formulae to determine the values of the parameters (Visioli, 2006a). Note also that the use of a first-order filter that makes the controller transfer function proper has been neglected for the sake of clarity but it can be easily selected so that it does not influence the PID controller dynamics significantly.
We assume that the PID controller has been (roughly) tuned and a step signal of amplitude A_s is applied to the set-point. The process gain μ can be determined by considering the following trivial relations which involve the final steady-state value of the control variable u and of the control error e:

$$\lim_{t\to+\infty} u(t) = \frac{K_p}{T_i}\int_0^\infty e(t)dt = \frac{A_s}{\mu} \tag{5}$$

and therefore we have

$$\mu = A_s\frac{T_i}{K_p\int_0^\infty e(t)dt}. \tag{6}$$

The determination of the sum of the time constants T_0 of the process can be performed by considering the following variable:

$$e_u(t) = \mu u(t) - y(t). \tag{7}$$

By applying the Laplace transform to (7) and by expressing u and y in terms of the reference signal r we have

$$E_u(s) = \mu U(s) - Y(s) = \frac{C(s)(\mu - P(s))}{1 + C(s)P(s)}R(s). \tag{8}$$

At this point, for the sake of clarity, it is convenient to write the controller and process transfer functions respectively as

$$C(s) = \frac{K_p}{T_i s}\tilde{C}(s) \tag{9}$$

where

$$\tilde{C}(s) := (T_i s + 1)(T_d s + 1) \tag{10}$$

and (see (3))

$$P(s) = \frac{\mu}{q(s)} \tag{11}$$

where

$$q(s) = \prod_i(\tau_i s + 1) = \prod_i \tau_i s^n + \cdots + \sum_i \tau_i s + 1 \tag{12}$$

with

$$\tau_i := \left[\frac{2}{(2i+1)\pi}\right]^2 \tau, \qquad i = 0, 1, \ldots. \tag{13}$$

Then, expression (8) can be rewritten as

$$E_u(s) = \frac{\mu K_p\tilde{C}(s)}{T_i s q(s) + \mu K_p\tilde{C}(s)}(q(s) - 1)R(s). \tag{14}$$

By applying the final value theorem to the integral of e_u when a step is applied to the set-point signal we finally obtain (see (12))

$$\begin{aligned}\lim_{t\to+\infty}\int_0^t e_u(v)dv &= \lim_{s\to 0} s\frac{A_s}{s}\frac{\mu K_p\tilde{C}(s)}{T_i s q(s) + \mu K_p\tilde{C}(s)}\frac{q(s)-1}{s} \\ &= A_s\lim_{s\to 0}\frac{q(s)-1}{s} \\ &= A_s\sum_i \tau_i \\ &= A_s T_0.\end{aligned} \tag{15}$$

Thus, the sum of the time constants of the process can be obtained by evaluating the integral of $e_u(t)$ at the steady-state when a step signal is applied to the set-point and by dividing it by the amplitude A_s of the step.
Remark 1. It is worth noting that both the value of the gain μ and of sum of the time constants T_0 of the process are determined by considering the integral of signals and therefore the method is inherently robust to the measurement noise.
Remark 2. Note also that the set-point step signal can be applied just for the purpose of (re)tuning the PID

controller (in this case its amplitude should be as small as possible in order perturb the process as less as possible) but also a step response during routine process operations can be employed. This issue will be further discussed in subsection 3.5.

Remark 3. It is worth stressing that the value of T_0 is obtained independently on the values of the PID parameters. This is an advantage with respect to the use of other methods for the identification of the process transfer function, whose result depends on the control variable and process variable signals.

3.2 Estimation of the process parameters from load disturbance step response

The process parameters can be estimated by evaluating also a load disturbance step d of amplitude A_d. However, in this case the amplitude A_d is not known and therefore must be estimated as well. This can be determined by considering the final value of the integral of the control error. In fact, the expression of the Laplace transform of the control error is:

$$E(s) = -\frac{P(s)}{1+C(s)P(s)}D(s) = -\frac{T_i s\mu}{T_i s q(s) + K_p \tilde{C}(s)\mu}\frac{A_d}{s}, \tag{16}$$

and therefore we obtain

$$\begin{aligned}\lim_{t\to+\infty}\int_0^t e(v)dv &= \lim_{s\to 0} s\frac{1}{s}\frac{A_d}{s}\left(-\frac{T_i s\mu}{T_i s q(s) + K_p\tilde{C}(s)\mu}\right)\\ &= -\frac{A_d T_i}{K_p}.\end{aligned} \tag{17}$$

Thus, the amplitude of the step disturbance can be determined as

$$A_d = -\frac{K_p}{T_i}\int_0^\infty e(t)dt. \tag{18}$$

Once the amplitude of the step disturbance has been determined, the process gain μ can be determined by first considering the Laplace transform of the process input $i = u + d$, that is:

$$\begin{aligned}I(s) &= U(s) + D(s)\\ &= -\frac{C(s)P(s)}{1+C(s)P(s)}D(s) + D(s)\\ &= \frac{1}{1+C(s)P(s)}\frac{A_d}{s}\\ &= \frac{T_i s q(s)}{T_i s q(s) + K_p\tilde{C}(s)\mu}\frac{A_d}{s}.\end{aligned} \tag{19}$$

Thus, if we integrate $i(t)$ and we determine the limit for $t \to +\infty$ we obtain

$$\lim_{t\to+\infty}\int_0^t i(v)dv = \lim_{s\to 0} s\frac{1}{s}\frac{T_i s q(s)}{T_i s q(s) + K_p\tilde{C}(s)\mu}\frac{A_d}{s} = \frac{T_i A_d}{\mu K_p} \tag{20}$$

The process gain μ can be therefore found easily, once the value of A_d has been determined by using (18), as

$$\mu = A_d\frac{T_i}{K_p\int_0^\infty (u(t)+A_d)dt}. \tag{21}$$

Finally, the determination of the sum of the time constants of the process can be performed by initially considering the variable

Table 1. Tuning rules for distributed-lag processes.

	K_p	T_i	T_d
PI	$5/\mu$	$0.54T_0$	0
PID	$100/15/\mu$	$0.25T_0$	$0.10T_0$

$$e_i(t) := \mu(u(t) + d(t)) - y(t). \tag{22}$$

By applying the Laplace transform to (22) and by expressing u and y in terms of d we have

$$\begin{aligned}E_i(s) &= \frac{\mu - P(s)}{1+C(s)P(s)}D(s)\\ &= \frac{\mu - P(s)}{1+C(s)P(s)}\frac{A_d}{s}\\ &= \frac{\mu T_i A_d s}{T_i s q(s) + K_p\mu\tilde{c}(s)}\frac{q(s)-1}{s}.\end{aligned} \tag{23}$$

By twice integrating e_i and by applying the final value theorem we obtain (see (15))

$$\begin{aligned}&\lim_{t\to+\infty}\int_0^t\int_0^{v_2} e_i(v_1)dv_1dv_2\\ &= \lim_{s\to 0} s\frac{1}{s^2}\frac{\mu T_i A_d s}{T_i s q(s) + \mu K_p\tilde{C}(s)}\frac{q(s)-1}{s}\\ &= \frac{T_i A_d}{K_p}T_0.\end{aligned} \tag{24}$$

Thus, T_0 can be obtained as

$$T_0 = \frac{K_p}{T_i A_d}\int_0^\infty\int_0^t e_i(v)dvdt. \tag{25}$$

Remark 4. Note that also in this case the estimation of the process parameters is based on the integral of signals and therefore the method is inherently robust to the measurement noise. Further, the process parameters are obtained independently on the values of the PID parameters, because the estimation is based on steady-state values of the variables. Finally, as for the set-point step response, the step disturbance signal can be applied just for the purpose of (re)tuning the PID controller (in this case its amplitude should be as small as possible in order to perturb the process as less as possible) but also a step response during routine process operations can be employed.

Remark 5. In the proposed method, the occurrence of an abrupt (namely, step-like) load disturbance has been assumed. Indeed, this is the most relevant case for the control system, as the disturbance excites significantly the dynamics of the control system itself. Thus, the performance assessment technique has to be implemented together with a procedure for the detection of abrupt load disturbances. Methods for this purpose have been proposed in (Hägglund and Åström, 2000; Veronesi and Visioli, 2008).

3.3 Tuning of the controller

Once the sum of the time constants has been estimated by evaluating the set-point or the load disturbance step response, the PID controller can be tuned properly by considering the load disturbance rejection task, which is usually of main concern in practical applications. We propose to use the tuning rules devised by Shinskey and explained in (Shinskey, 1994, 2001). They are reported in Table 1, for the sake of clarity, for both PI and PID controller.

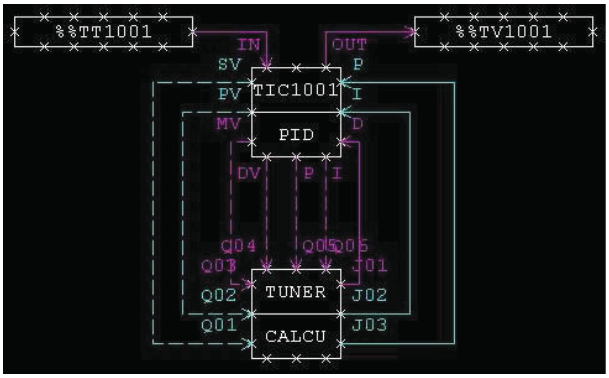


Fig. 3. The implementation of the proposed technique by means of the Yokogawa Centum VP Distributed Control System (courtesy of Yokogawa Italia).

3.4 Practical implementation

If a set-point step response is employed for estimating the process parameters, the proposed methodology can be easily implemented in a DCS with a suitable software development environment, as shown in Figure 3. The PID block (TIC1001) executes the standard PID control: its input and output are indicated respectively as TT1001 and TV1001. The calculation block TUNER determines the value of T_0 by computing the integral of the process variable (PV) and of the control output (MV) it receives from the PID block. Further, it computes the process gain. For this reason it needs also the setpoint (SV) and the PI parameters, namely, the proportional gain (or, equivalently, the proportional band) and the integral time constant. Finally the block TUNER computes the new values of the PID parameters by implementing the tuning rules shown in Table 1 and send them back to the PID block. Note that Q01..08 and J01..03 are the conventional name of the ports that the calculation block uses for exchanging data with the other function blocks.
If a load disturbance response is employed, the estimation procedure has to be estimate first the step amplitude A_d and then its value has to be employed to determine the μ and T_0 as indicated in (21) and (25).

3.5 Performance assessment

In a practical context it is also useful to evaluate the performance of a (PID) controller in order to determine if it has to be retuned or not. This is especially necessary if a self-tuning procedure has to be implemented, namely, the control system itself evaluates the control performance during process routine operations and a new tuning is provided in case it is not satisfactory. In this context, a measure of the performance of a control system can be effectively based on the integrated absolute error

$$IAE = \int_0^\infty |e(t)|dt \tag{26}$$

which implicitly considers both the peak error value and the settling time. For the technique proposed in this paper it is of interest to assess the control performance when a load disturbance occurs. For this purpose, the integrated absolute error obtained by applying the tuning rules of Table 1 to distributed-lag processes (2) with different values of μ and τ, and different process order n has been computed. Results for $\mu = 1$ are shown in Figure 4 and 5 for PI and PID controller respectively. By interpolating these results, we obtain that the value of IAE achieved by applying the tuning rules of Table 1 are (for PI and PID controllers respectively):

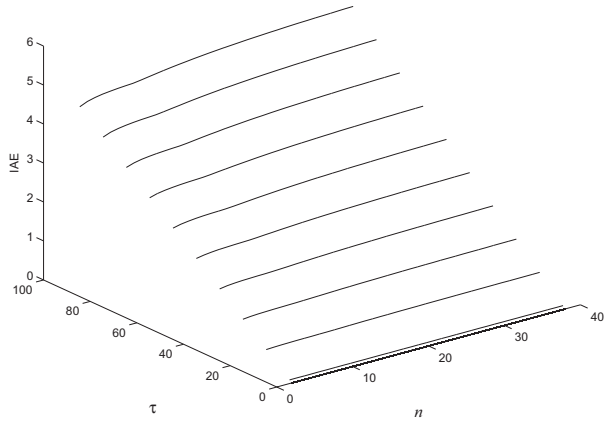


Fig. 4. Values of IAE for different values of τ and n (process order) with a PI controller tuned according to Table 1.

$$IAE_{PI} = 0.058\tau\mu = 0.116T_0\mu \tag{27}$$

$$IAE_{PID} = 0.02\tau\mu = 0.04T_0\mu \tag{28}$$

Thus, the integrated absolute error achieved by a PI(D) controller should be ideally that expressed in (27) and (28). A performance index can be therefore defined as

$$J_{PI} = \frac{IAE_{PI}}{\int_0^\infty |e(t)|dt} \tag{29}$$

$$J_{PID} = \frac{IAE_{PID}}{\int_0^\infty |e(t)|dt} \tag{30}$$

and it can be determined, once the process parameters have been estimated by applying the technique described previously, by considering the obtained integrated absolute error.
In principle, the performance obtained by the control system is considered to be satisfactory if $J_{PI} = 1$ or $J_{PID} = 1$. From a practical point of view, however, the controller can be considered to be well-tuned if J_{PI} or J_{PID} is greater than a threshold (less than one) which can be selected by the user depending on how tight are its control specifications. In any case a sensible default value of 0.8 can be fixed.
Remark 6. It turns out from the presented results that using the derivative action allows to improve the performance significantly with respect to a PI controller.
Remark 7. It is worth noting that a performance index J greater than one can result because of the (small) interpolation error in determining (29) and (30) and because in any case the tuning formulae of Table 1 does not guarantee that the integrated absolute error is globally minimized.

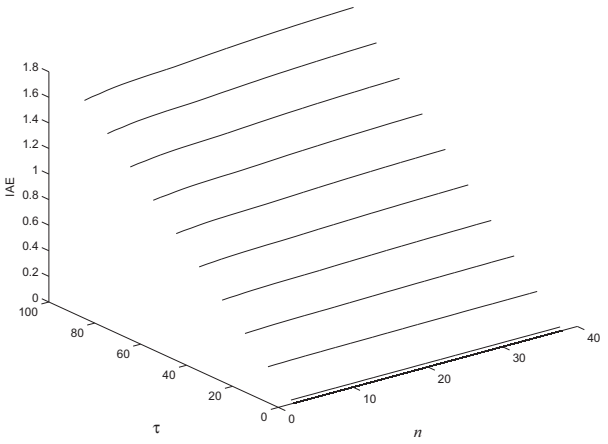


Fig. 5. Values of IAE for different values of τ and n (process order) with a PID controller tuned according to Table 1.

4. SIMULATION RESULTS

4.1 Example 1 - PID control

As a first example we consider a process with $\mu = 1$, $\tau = 10$ and $n = 30$ lags. Initially, the PID controller parameters are selected as $K_p = 3.3$, $T_i = 1.9$, $T_d = 0.25$. Then, a unit step load disturbance is applied to the process and the amplitude of the disturbance, the gain of the process and the sum of the time constants are estimated as $A_d = 1$, $\mu = 1.0$, and $T_0 = 4.97$. Based on these values, the PID parameters are retuned, according to Table 1, as $K_p = 6.66$, $T_i = 1.24$, $T_d = 0.5$. The load disturbance step response provided by the new values of the PID controller parameters is shown as a solid line in Figure 6, where the load disturbance step response provided by the initial values is also plotted as a dashed line. The control signal is not shown for the sake of brevity, in any case there are no significant differences between the two cases. By retuning the controller, the performance index is improved from $J_{PID} = 0.32$ to $J_{PID} = 1.03$ while the integrated absolute error decreases from IAE = 0.62 to IAE = 0.19. It is worth noting that the same result is achieved if a set-point step response is employed for estimating the process parameters.

4.2 Example 2 - PI control

As a second example we consider the same process of Example 1, but the use of a PI controller is assumed. Initially, the controller parameters are selected as $K_p = 7$ and $T_i = 2$ (note that the controller is aggressive). Then, a unit step load disturbance is applied to the process and the amplitude of the disturbance, the gain of the process and the sum of the time constants are estimated as $A_d = 1$, $\mu = 0.99$, and $T_0 = 4.97$ (the same parameters are estimated by considering a set-point step response). Based on these values, the PI parameters are retuned, according to Table 1, as $K_p = 5.06$ and $T_i = 2.68$. The load disturbance step responses provided by the initial and new values of the PI controller parameters are shown in Figure 7 as a dashed and solid line respectively. As

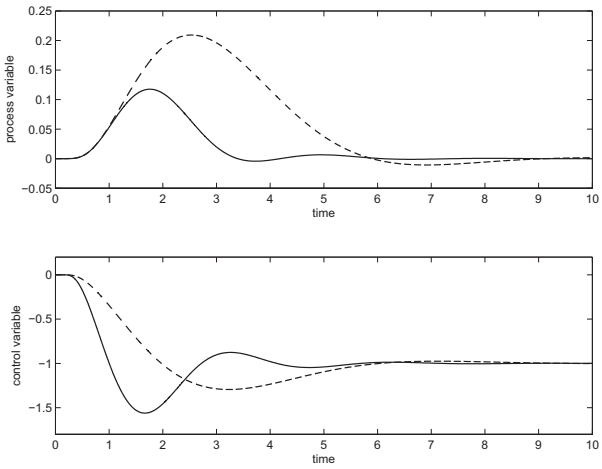


Fig. 6. Load disturbance step response for example 1. Dashed line: initial tuning. Solid line: automatic tuning.

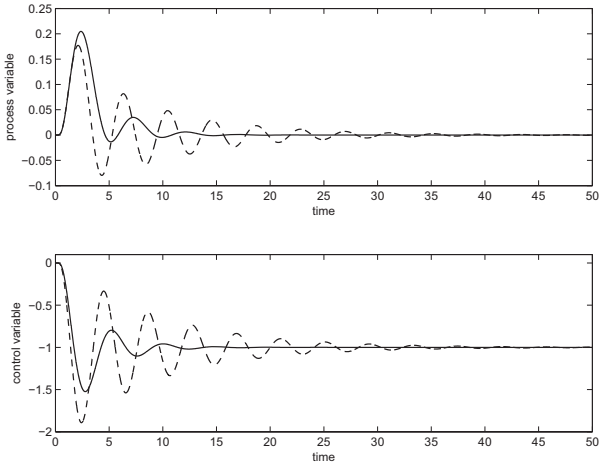


Fig. 7. Load disturbance step response for example 2. Dashed line: initial tuning. Solid line: automatic tuning.

in Example 1, retuning the controller allows to increase the performance. In particular, the performance index is improved from $J_{PI} = 0.64$ to $J_{PID} = 1.01$ while the integrated absolute error decreases from IAE = 0.88 to IAE = 0.56.

It turns out that the proposed autotuning method is effective and, by comparing these results with those of Example 1, it appears that the use of the derivative action allows to increase the controller performance significantly.

4.3 Example 3 - Measurement noise

As a third example we consider again the same process of Example 1, but the process output is corrupted with zero-mean white noise with a variance of $0.1 \cdot 10^{-3}$. The load disturbance step response obtained by selecting the controller parameters as $K_p = 3$, $T_i = 2$, and $T_d = 0.5$ is shown in Figure 8. In order to determine the performance index J_{PID} correctly, it is necessary to discard from the computation of the integrated absolute error those areas

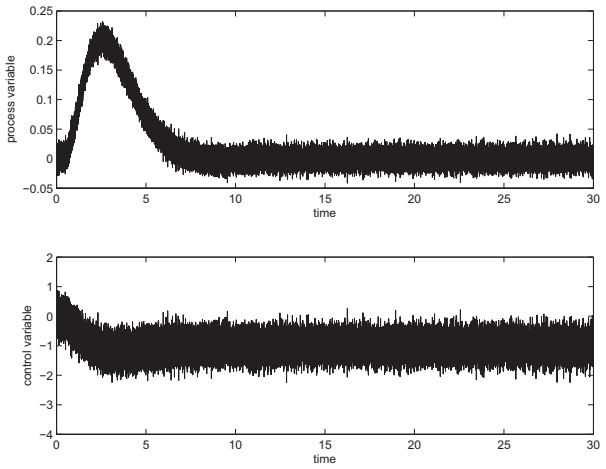


Fig. 8. Load disturbance step response for example 3 with $K_p = 3$, $T_i = 2$, and $T_d = 0.5$.

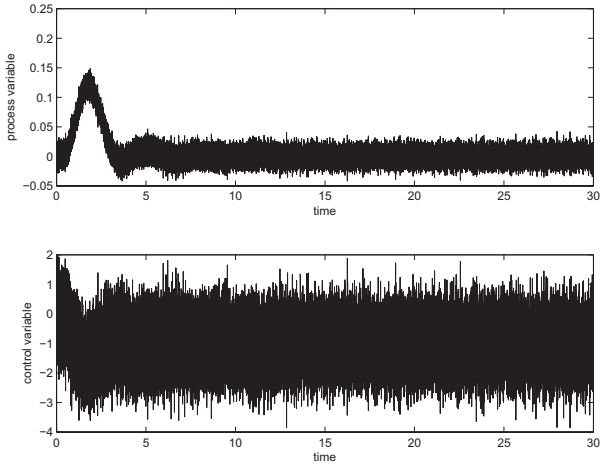


Fig. 9. Load disturbance step response for example 3 with $K_p = 6.87$, $T_i = 1.26$, and $T_d = 0.51$.

whose value is less than a predefined threshold (because they are actually due to the noise) (Visioli, 2006b). It results $J_{PID} = 0.29$, which suggests that the controller needs to be retuned. The gain of the process and the sum of the time constants are then estimated as $A_d = 1$, $\mu = 0.97$, and $T_0 = 5.06$ (once again, note that virtually the same values are obtained by considering a set-point step response). Based on these values, the PID parameters are retuned, according to Table 1, as $K_p = 6.87, T_i = 1.26$, and $T_d = 0.51$. The load disturbance step response obtained with the new PID controller is shown in Figure 9. In this case the performance index is $J_{PID} = 1.06$. By retuning the controller the integrated absolute error is decreased from IAE = 0.68 to IAE = 0.19. It turns out that the presence of noise does not impair the effectivess of the method, as expected because the considered variables are integrated.

5. CONCLUSIONS

In this paper we have proposed an automatic tuning methodology for distributed-lag processes based on a closed-loop experiment. Being based on the evaluation of a set-point or load disturbance step response, the technique can employ process routine operating data and can therefore be extended straightforwardly as a self-tuning method. Indeed, a performance index has been devised in order to assess the performance of the controller based on the achieved integrated absolute error. Illustrative examples have shown the effectiveness of the method and that it is robust to the measurement noise. Thus, the methodology appears to be suitable to implement in an industrial setting.

REFERENCES

T. Hägglund and K. J. Åström. Supervision of adaptive control algorithms. *Automatica*, 36(2):1171–1180, 2000.

A. O'Dwyer. *Handbook of PI and PID Tuning Rules*. Imperial College Press, 2006.

F. G. Shinskey. *Feedback Controllers for the Process Industries*. McGraw-Hill, New York, USA, 1994.

F. G. Shinskey. PID-deadtime control of distributed processes. *Control Engineering Practice*, 9(11):1177–1183, 2001.

F. G. Shinskey. Process control: as taught vs as practiced. *Industrial and Engineering Chemistry Research*, 41(16): 3745–3750, 2002.

M. Veronesi and A. Visioli. Performance assessment and retuning of pid controllers. *Industrial and Engineering Chemistry Research*, 48(5):2616–2623, 2009.

M. Veronesi and A. Visioli. A technique for abrupt load disturbance detection in process control systems. In *Proceedings 17th IFAC World Congress*, Seoul, ROK, 2008.

A. Visioli. *Practical PID Control*. Springer, London, UK, 2006a.

A. Visioli. Method for proportional-integral controller tuning assessment. *Industrial and Engineering Chemistry Research*, 45:2741–2747, 2006b.

C. C. Yu. *Autotuning of PID Controllers*. Springer-Verlag, London, Great Britain, 1999.

New tuning rules for PI and fractional PI controllers

Juan J. Gude and Evaristo Kahoraho

Faculty of Engineering – ESIDE
University of Deusto
Avda. de las Universidades 24, 48007 Bilbao (Spain)
(Tel: +34 944139000; Fax: +34 944139101; e-mail: jgude@eside.deusto.es)

Abstract: This paper presents new tuning rules for PI and fractional PI control of processes that are typically found in process control. The rules are based on characterization of process dynamics by three parameters that can be obtained from a simple step response experiment. The rules are obtained by minimizing a frequency objective function subject to a constraint on the maximum sensitivity. Comparisons with classical tuning rules show that they are very simple but give substantially better performance.

Keywords: Fractional control, PID control, design, tuning methods, optimization, process control.

1. INTRODUCTION

In spite of all the advances in process control over the past several decades, the proportional integral (PI) and the proportional integral derivative (PID) controller remains to be certainly the most extensive option that can be found on industrial control applications, see Åström and Hägglund (2001). The transparency of the PID control mechanism, the availability of a large number of reliable and cost-effective commercial PID modules, and their widespread acceptance by operators are among the reasons of its success, see Gude and Kahoraho (2007).

Over the last half-century, a great deal of academic and industrial effort has focused on improving PID control, primarily in the area of tuning rules. In fact, since Ziegler and Nichols proposed their popular tuning rules, Ziegler and Nichols (1942), an intensive research has been done. Works include from modifications of the original tuning rules, see Chien *et al.* (1952), Hang *et al*. (1991), and Åström and Hägglund (2004), to a variety of new techniques, see Åström and Hägglund (1995).

Fractional calculus, which is the expansion to fractional orders, has been known since the development of the regular calculus. However, fractional-order control was not incorporated into control engineering mainly due to the lack of sufficient mathematical knowledge and the limited computational power available at that time.

More recently, Podlubny (1999) has proposed a generalization of the PI and PID controllers, namely the PI^{λ} and $PI^{\lambda}D^{\mu}$ controllers, involving an integrator of order λ and a differentiator of order μ (the orders λ and μ may assume real non-integer values). Podlubny has also demonstrated the better response of these types of controllers, in comparison with the classical PI and PID controllers, when used for the control of fractional-order systems. A frequency domain approach by using fractional PID controllers has also been studied in Vinagre et al. (2000). However, the design methods for fractional controllers are a recent research area, see Capponetto *et al.* (2002) and Monje *et al.* (2005).

Given that the most common control structure used in the process industry is the PI controller, Åström and Hägglund (2001), an immediate approach that should be taken into account is to use the fractional PI^{λ} controller. Because of the widespread use of PI controllers and the potentials of fractional PI^{λ} controllers, see Gude and Kahoraho (2009), it is interesting to have simple but efficient methods for tuning these kind of controllers.

In this paper, we have developed new simple tuning methods for PI and PI^{λ} controllers that give significantly better performance for a wide range of processes.

The layout of this paper is the following. The different controllers and the test batch considered in this paper are presented in Section 2. The design method is treated in Section 3. This is followed by the main results obtained in this paper: new tuning rules for PI and PI^{λ} controllers in Section 4. In Section 5 the developed tuning rules are applied to a process and a comparison between different tuning rules is made. Finally conclusions and final remarks are drawn in Section 6.

2. CONTROLLERS AND TEST BATCH

2.1 Plant knowledge

To be accepted in industrial applications controller tuning rules must be based on a limited amount of plant knowledge that is easy to obtain. The plant can the be characterized by its τ value:

$$\tau = \frac{L}{L+T} \quad (1)$$

This parameter is usually called the *normalized dead time*. It is essentially the classical *controllability ratio* L/T, but the parameter τ has the advantage that it is in the range from 0 to 1. The *controllability ratio* was often mentioned in the early process control literature, see Cohen and Coon (1953). This parameter can be used to characterize the difficulty of controlling a process. Roughly speaking, processes with small τ can be considered easy to control and the difficulty in controlling the system increases as τ increases.

2.2 The test batch

The design method presented in the next section requires the transfer function of the process to be known. The results of this investigation depend critically on the chosen test batch. To apply the method we therefore have to choose process models that are representative for the dynamics of typical industrial processes. Processes with the following transfer functions have been used:

$$G_1(s) = \frac{e^{-s}}{(1+sT)^2}$$

T = 0.01, 0.05, 0.1, 0.2, 0.3, 0.5, 0.7, 1, 2, 4, 6, 8, 10

$$G_2(s) = \frac{1}{(s+1)^n}$$

n = 3, 4, 5, 6, 7, 8

$$G_3(s) = \frac{1}{(1+s)(1+\alpha s)(1+\alpha^2 s)(1+\alpha^3 s)} \quad (2)$$

α = 0.1, 0.2, 0.5, 0.7

$$G_4(s) = \frac{1-\alpha s}{(s+1)^3}$$

α = 0.1, 0.2, 0.5, 1, 2

$$G_5(s) = \frac{1}{(1+s)(1+sT)}$$

T = 0.02, 0.05, 0.1, 0.2, 0.5

The process (3) is the standard model that has been used in many investigations of PID tuning.

$$G(s) = K_P \frac{e^{-Ls}}{(1+sT)} \quad (3)$$

The test batch (2) does, however, not include this transfer function because this model is not representative for typical industrial processes, see Åström and Hägglund (1995). Tuning based on the model (3) typically gives controller gains that have a different behaviour from the other processes in the test batch, see Hang *et al.* (1991). This is remarkable because tuning rules have traditionally been based on this model.

The processes selected in the test batch (2) are representative for many of the processes typically found in process control, see for example Åström and Hägglund (2000) and Gorez (2003), suggested as standard benchmark models for testing PID controllers. The test batch includes processes that range from delay-dominated to lag-dominated processes. They include all kinds of plants with poles strictly on the negative real axis, such as plants with time delay or non-minimum phase zeros, plants of high and low orders, plants with multiple and spread poles, etc. All processes are normalized to have unit steady state gain and have a parameter that can be changed to influence the response of the process. The parameter ranges have been chosen to give a wide variety of responses. The normalized time delay ranges from 0.17 to 1 for G_1. The rest of the processes have values of τ in the range $0 < \tau < 0.5$

2.3 PI and PI$^\lambda$ controllers

In this paper, two different controllers are considered: the PI and the fractional PI$^\lambda$ controller, which is a generalisation of the PI controller. It is a non-integer order controller of the form:

$$C(s) = K + \frac{k_i}{s^\lambda} \quad (4)$$

where K is the proportional gain, k_i the integral gain, and λ the fractional order of the integral part.

The interest of this kind of controller is justified by a better flexibility, since it exhibits a fractional integral part of order λ. Thus, three parameters can be tuned in this structure (K, k_i, and λ), that is, one more parameter than in the case of conventional PI controller (λ = 1). We can take advantage of the fractional order λ to improve the performance.

3. THE DESIGN METHOD

Within the process industry, regulation performance is often of primary importance since most controllers operate as regulators, see Shinskey (1996). Regulation performance is often expressed in terms of the control error obtained for certain disturbances. A load disturbance is typically applied at the process input. Typical criteria are to minimize a loss function of the form:

$$I = \int_0^\infty t^n |e(t)|^m dt \quad (5)$$

where the error is defined as e(t) = r(t) – y(t). Common cases are IAE (n = 0, m = 1), ISE (n = 0, m = 2), or ITSE (n = 1, m = 2).

However, Kristiansson and Lennartson (2002) defined another performance criterion in the frequency domain as an alternative to the above criteria based on a function of the error signal. It is formulated as:

$$J_v = \left\| \frac{1}{s} G(s) S(s) \right\|_\infty = \max_\omega \left| \frac{1}{j\omega} \cdot \frac{G(j\omega)}{1+L(j\omega)} \right| \quad (6)$$

The proposed performance criterion is mainly a measure of the system ability to handle low-frequency load disturbances.

Robustness is an important consideration in control design. There are many different criteria for robustness. Many of them can be expressed as restrictions on the Nyquist curve of

the loop transfer function L(s) = G(s)C(s). Åström and Hägglund (1995) introduced the maximum sensitivity function of the closed-loop system, M_S, as a tuning parameter for PID controllers. The constraint (7) that sensitivity function S(jω) is less than a given value M_S implies that the loop transfer function should be outside a circle with radius $1/M_S$ and center at –1.

$$\|S(s)\|_\infty = \max_\omega |S(j\omega)| = \max_\omega \left|\frac{1}{1+L(j\omega)}\right| \le M_S \qquad (7)$$

The design problem discussed in this paper can be formulated as an optimisation problem: *Find parameters of the different controllers that minimize performance criterion (6) subject to the robustness constraint (7).*

A reasonable ambition in all control design is to keep the control signal as small as possible. Control system design very often deals with the trade-off between performance and control effort, provided that a reasonable mid-frequency robustness is guaranteed, see for example Gude and Kahoraho (2009). Therefore, introduce the control effort criterion:

$$J_u = \|C(s)S(s)\|_\infty = \max_\omega \left|\frac{C(j\omega)}{1+L(j\omega)}\right| \qquad (8)$$

4. RESULTS

An empirical method is used to develop the new tuning rules. The design method proposed in Section 3 with M_S = 1.4 was applied to all processes in the test batch (2). This value of M_S provides a good compromise between performance and robustness. This gave the corresponding parameters K, T_i, for the PI, and K, T_i, and λ, for the fractional PI controller. The process parameters K_P, L and T were also computed from the step response experiment. The controller gain is normalized by multiplying it either with the static process gain K_P or with the parameter a = K_PL/T. Integration time is normalized by dividing by T or by L. We will represent normalized controller parameters as functions of τ. Data obtained can be well approximated by functions having the form:

$$f(\tau) = a\tau^b + c \qquad (9)$$

4.1 PI controller

Simplified tuning rules for PI controllers will be first obtained. Figures 1 and 2 show the normalized proportional gains and integration times, respectively, as a function of normalized time delay τ when the design procedure is applied to all processes in the test batch (2). The curves drawn correspond to the results obtained by curve fitting. Both figures show that there appears to be a good correlation between the normalized controller parameters and the normalized time delay τ. This indicates that it is possible to develop good tuning rules based on the *KLT*-model. However, Figures 1 and 2 also show that parameters KK_P, aK, T_i/L, and T_i/T range from 0.16 to 23.8, from 0.21 to 3.15, from 0.34 to 8.2, and from 0.1 to 6.8, respectively.

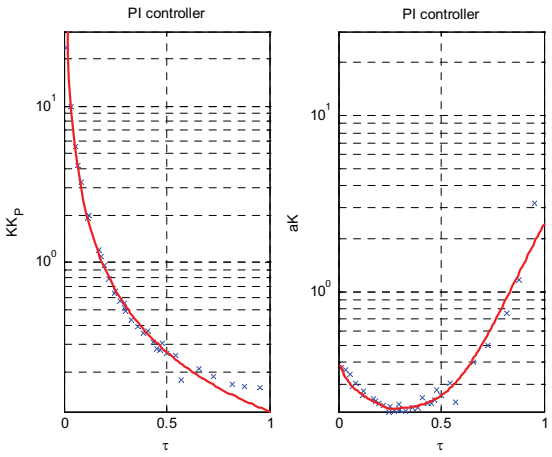


Fig. 1. Normalized PI controller proportional gains plotted versus normalized time delay τ for the test batch. The solid lines correspond to the tuning rules obtained in Table 1.

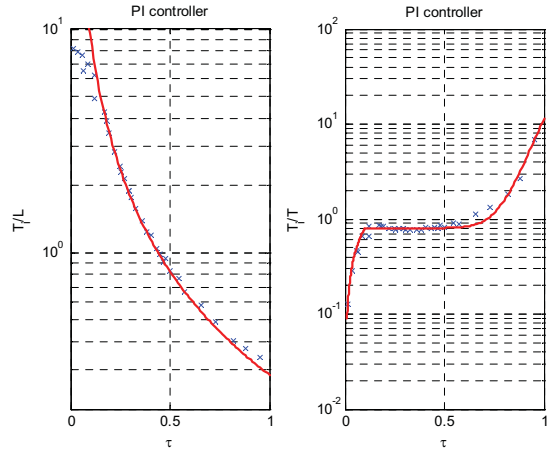


Fig. 2. Normalized PI controller integration times plotted versus normalized time delay τ for the test batch. The solid lines correspond to the tuning rules obtained in Table 1.

This indicates clearly that it is not possible to obtain good tuning rules that do not depend on τ. The deviations from the solid lines in the figure is about ± 15%

Table 1. Tuning formulae for the PI controller. The table gives the parameters of the functions of the form (9) for the normalized controller parameters and M_S = 1.4.

f(τ)	a	b	c	τ
KK_P	0.09793	–1.3676	0.01378	0 < τ < 1
aK	–0.6473	0.1128	0.77	0 < τ < 0.25
	2.212	5.7	0.2163	0.25 < τ < 1
T_i/L	0.2967	–1.497	–0.01252	0 < τ < 1
T_i/T	5.479	0.8154	–0.03853	0 < τ < 0.1
	10.7	11.79	0.8028	0.1 < τ < 1

Table 1 gives the coefficients for functions of the form (9) fitted to the data available in Figures 1 and 2. The corresponding graphs are shown in solid lines in figures.

4.2 PI^{λ} controller

Simplified tuning rules for fractional PI controllers will be now obtained. Figures 3, 4, and 5 show the normalized proportional gains, the normalized integration times, and the controller fractional order, respectively, as a function of normalized time delay τ when the design procedure is applied to all processes in the test batch (2). The curves drawn correspond to the results obtained by curve fitting. Both figures show that there appears to be a good correlation between the normalized controller parameters and the normalized time delay τ. This indicates that it is possible to develop good tuning rules for fractional PI controllers based on the *KLT*-model.

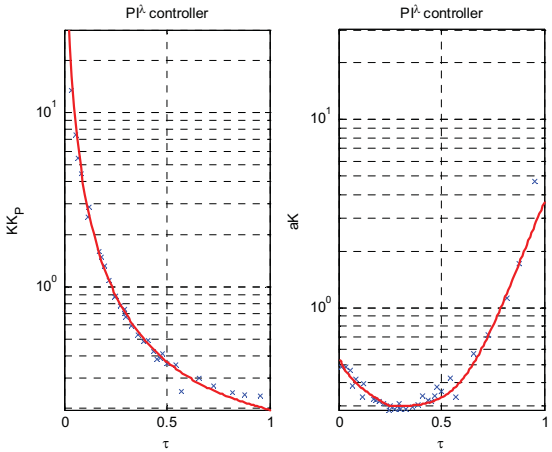


Fig. 3. Normalized PI^{λ} controller proportional gains plotted versus normalized time delay τ for the test batch. The solid lines correspond to the tuning rules obtained in Table 2.

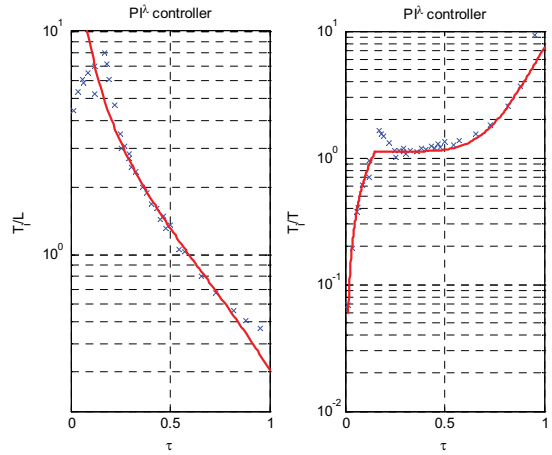


Fig. 4. Normalized PI^{λ} controller integration times plotted versus normalized time delay τ for the test batch. The solid lines correspond to the tuning rules obtained in Table 2.

As in the case of the PI controller, it is not possible to obtain good tuning rules that do not depend on τ. The deviations from the solid lines in the figure is about ± 15%. Table 2 gives the coefficients of functions of the form (9) fitted to the data available in Figures 3, 4, and 5. The corresponding graphs are shown in solid lines in these figures.

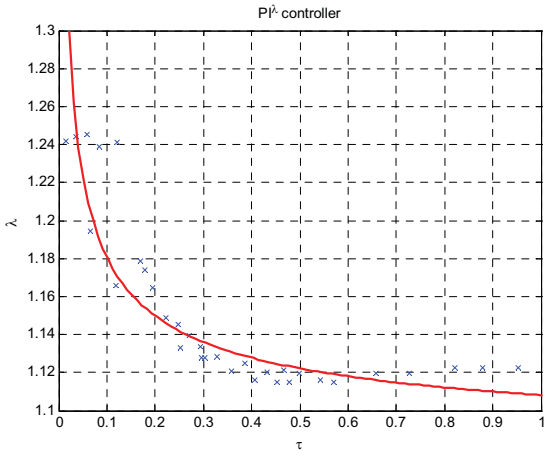


Fig. 5. Fractional order λ plotted versus normalized time delay τ for the test batch. The solid lines correspond to the tuning rules obtained in Table 2.

Table 2. Tuning formulae for the fractional PI^{λ}. The table gives the parameters of the functions of the form (9) for the normalized controller parameters and $M_S = 1.4$.

f(τ)	a	b	c	τ
KK_P	0.08621	–1.594	0.1096	$0 < \tau < 1$
aK	–0.5643	0.2715	0.6866	$0 < \tau < 0.25$
	3.327	6.593	0.2983	$0.25 < \tau < 1$
T_i/L	1.17	–0.8997	–0.8666	$0 < \tau < 1$
T_i/T	8.549	1.052	–0.04380	$0 < \tau < 0.15$
	6.271	7.304	1.12	$0.15 < \tau < 1$
λ	0.03512	–0.4862	1.073	$0 < \tau < 1$

Figure 6 shows the ratio between the optimal J_v–values obtained with a PI^{λ} and PI controller applied to the processes in the test batch. It shows that the benefit in using a PI^{λ} instead a PI controller is more than 12% for delay–dominated processes, about 11% for balanced lag and delay processes, and tends to 18% for lag–dominated processes.

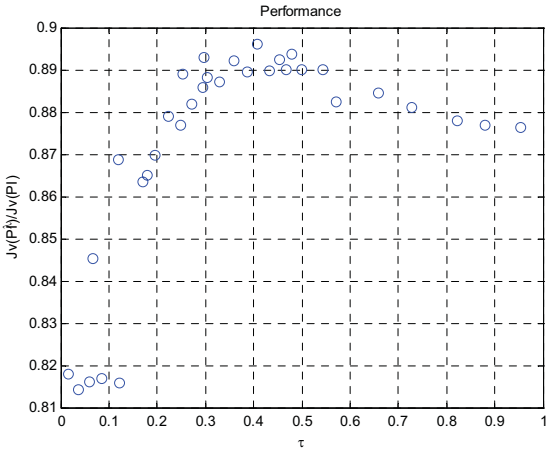


Fig. 6. Ratio of the J_v–values obtained for a PI^{λ} and a PI controller for different values of τ applied to the processes of the test batch.

4.3 A simpler tuning rule for PI$^{\lambda}$ controllers

As can be seen in Figure 5, optimal λ–value is approximately equal to 1.12 for $0.3 < \tau < 1$. Provided that the maximum difference between the optimal value of λ and 1.12 is, in the worst case, equal to 0.12, i.e. 10%, we will try to develop simple tuning rules for PI$^{\lambda}$ controllers, fixing the value of λ to 1.12. Figures 7 and 8 show the optimal normalized proportional gains and integration, for a constant value of $\lambda = 1.12$, as a function of the normalized time delay τ. The curves drawn correspond to the results obtained by curve fitting in Table 3.

Figure 9 shows the ratio between the optimal J_v–values obtained with a PI$^{\lambda}$ with all its parameters free and a PI$^{\lambda}$ with $\lambda = 1.12$ applied to the test batch. It shows that the J_v–values obtained in both cases are nearly the same for $0.3 < \tau < 1$, and the loss for lag–dominated processes increases but it is, in all cases, less than 5%.

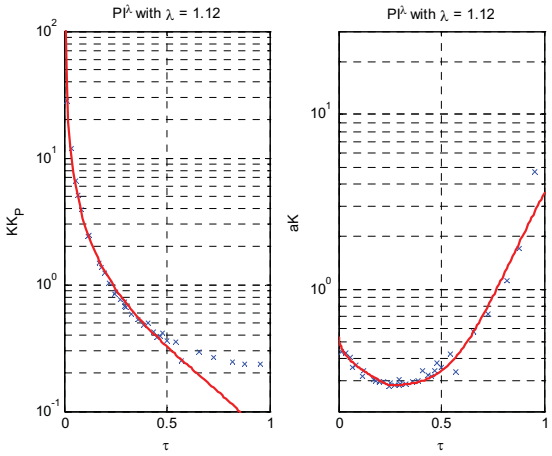


Fig. 7. Normalized PI$^{\lambda}$ controller proportional gains plotted versus normalized time delay τ for the test batch. The solid lines correspond to the tuning rules obtained in Table 3.

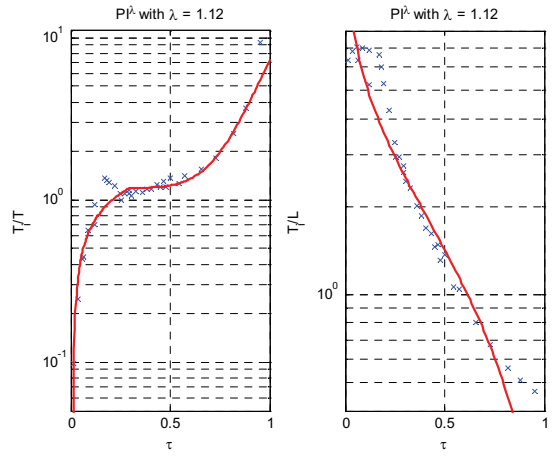


Fig. 8. Normalized PI$^{\lambda}$ controller integration times plotted versus normalized time delay τ for the test batch. The solid lines correspond to the tuning rules obtained in Table 3.

Table 3. Tuning formulae for fractional PI$^{\lambda}$ controllers. The table gives the parameters of the functions of the form (9) for the normalized controller parameters, $\lambda = 1.12$ and $M_S = 1.4$.

f(τ)	a	b	c	τ
KK_P	0.2154	–1.169	–0.1592	$0 < \tau < 1$
aK	–0.4645	0.3182	0.5795	$0 < \tau < 0.25$
	3.271	5.75	0.28	$0.25 < \tau < 1$
T_i/L	9.242	–0.1966	–9.171	$0 < \tau < 1$
T_i/T	5.479	0.8154	–0.03853	$0 < \tau < 0.3$
	6.06	7.066	1.18	$0.3 < \tau < 1$
λ	$\lambda = 1.12$			$0 < \tau < 1$

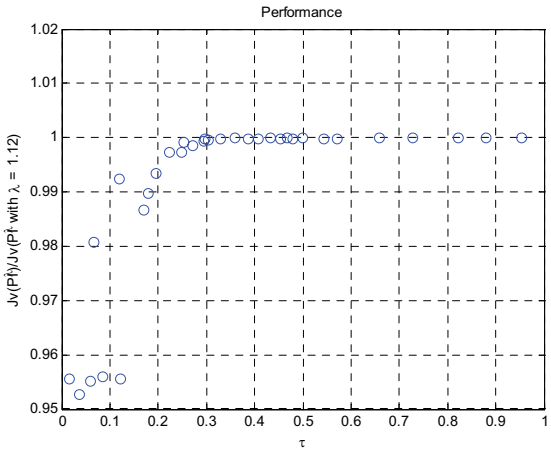


Fig. 9. Ratio of the J_v–values obtained for a PI$^{\lambda}$ and a PI$^{\lambda}$ with $\lambda = 1.12$ for different values of τ applied to the processes of the test batch.

5. COMPARISON WITH OTHER DESIGN METHODS

Extensive simulations have been done. Comparisons with classical tuning rules show that proposed tuning rules are very simple but give substantially better performance. However, due to page limitations, only one simulation has been included in this paper. There are many methods for tuning PI controllers. In this Section, the proposed methods for PI and PI$^{\lambda}$ controllers are compared with the Ziegler-Nichols step response method, Ziegler and Nichols (1942), the Cohen-Coon method, Cohen and Coon (1953), and optimal controllers in terms of J_v, IAE, J_u, and M_S.

For simplicity we will denote the Ziegler Nichols step response method by ZN, the Cohen-Coon method by CC, the optimal PI and PI$^{\lambda}$ controller obtained using the design method by opt-PI and opt-PI$^{\lambda}$, respectively, the proposed method for PI controllers by GK, the one for PI$^{\lambda}$ controllers by f-GK, and the approximation for PI$^{\lambda}$ controllers with $\lambda = 1.12$ by af-GK.

Consider the process with the following transfer function: $G(s) = 1/(1+s)(1+0.5s)$. We find that the apparent time delay and time constants are $L = 0.193$ and $T = 1.407$. Hence, the controllability index is $L/T = 0.1372$ and $\tau = 0.12$ for this process.

Table 4 contains the values of the different controller parameters obtained with the considered design methods. This table shows that results obtained by GK, f-GK and af-GK are very close to their respective optimal values. The performance obtained for PI^{λ} is substantially better than for PI. ZN and CC give controllers that reduce load disturbances very effectively, however they exhibit a very poor robustness and excessively large control effort.

Table 4. Controller parameters obtained for the different design methods for the considered transfer function.

Method	K	T_i	λ	k_i	M_S	J_u	J_v	IAE
ZN	6.56	0.58	1	11.34	3.11	22.59	0.12	0.16
CC	6.65	0.50	1	13.32	3.62	28.90	0.14	0.18
GK	1.78	1.13	1	1.58	1.38	2.65	0.63	0.63
opt-PI	2.04	1.21	1	1.68	1.40	3.02	0.59	0.59
f-GK	2.62	1.24	1.17	2.12	1.41	3.58	0.45	0.58
opt-PI^{λ}	2.86	1.34	1.24	2.13	1.40	3.79	0.46	0.61
af-GK	2.40	1.32	1.12	1.82	1.39	3.28	0.50	0.60

Parameters obtained by f-GK and optimal PI^{λ} are very close which indicates that little is lost by not using the full transfer function. The improvement in J_v of using a PI^{λ} instead of a PI is about 22%. The value of J_v for GK is about 28% higher compared with f-GK. These improvements are also evaluated in Gude and Kahoraho (2009).

6. CONCLUSIONS

This paper presents new tuning rules for PI and fractional PI control of typical processes found in process control. The rules are based on characterization of the process dynamics by three parameters, i.e. gain K_P, apparent time constant T and apparent time delay L, that can be obtained by a simple step response experiment. The design method consists on minimizing a frequency objective function subject to a constraint on the maximum sensitivity function. Based on these parameters it is possible to develop very simple tuning rules for PI and PI^{λ} controllers that only depend on the normalized time delay τ.

In this paper it is also demonstrated that substantially better performance can be obtained using PI^{λ} instead of PI controllers. These tuning rules are shown to give good results compared to a couple of well established classical tuning methods, especially when simplicity, performance and robustness are emphasized.

Future investigation should rely on extending these tuning rules to fractional PID controllers.

ACKNOWLEDGEMENTS

This paper has mainly been written during a stay of the first author at the Laboratory IMS Bordeaux (France) from June to October 2008, which has been supported by DEIKER – *Agencia para la Promoción y Gestión de la Investigación*, University of Deusto.

REFERENCES

Åström, K.J. and Hägglund, T. (1995). *PID Controllers: Theory, Design, and Tuning.* Instrument Society of America. Research Triangle Park, NC.

Åström, K.J. and Hägglund, T. (2000). Benchmark systems for PID control. *Proceedings of IFAC workshop on Digital Control*, pp. 165-166. Terrassa, Spain.

Åström, K.J. and Hägglund, T. (2001). The future of PID control. *Control Engineering Practice*, Vol. 9 (11), pp. 1163-1175.

Åström, K.J. and Hägglund, T. (2004). Revisiting the Ziegler-Nichols step response method for PID control. *Journal of Process Control*, Vol. 14(6), pp. 635-650.

Capponetto, R., Fortuna, L., and Porto, D. (2002). Parameter tuning of a non integer order PID controller. *Proceedings of 15th Int. Symp. on Mathematical Theory of Networks and Systems*. Notre Dame, IN.

Chien, K.L., Hrones, J.A. and Reswick, J.B. (1952) On the automatic control of generalized passive systems, *Transactions ASME*, Vol. 74, pp. 175-185.

Cohen, G.H. and Coon, G.A. (1953). Theoretical consideration of retarded control, *Transactions ASME*, Vol. 75, pp. 827-834.

Gorez, R. (2003). New design relations for 2-DOF PID-like control systems. *Automatica*, Vol. 39 (5), pp. 901-908.

Gude, J.J. and Kahoraho, E. (2007). PID control: Current status and alternatives. *In 2nd Seminar for Advanced Industrial Control Applications – SAICA 2007*, pp. 247-253. UNED Ediciones.

Gude, J.J. and Kahoraho, E. (2009). Performance comparison between PI(D) and fractional PI controllers. *In IFAC Workshop on Control Applications and Optimization – CAO'09*. University of Jyvaskyla, Finland.

Hang, C.C., Åström, K.J., and Hägglund, T. (1991). Refinements of the Ziegler-Nichols tuning formula. *IEE Proc., Part D,* Vol. 138 (2), pp. 111-118.

Kristiansson, B. and Lennartson, B. (2002). Robust and optimal tuning of PI and PID controllers. *IEE Proc., Part D*, Vol. 149 (1), pp. 17-25.

Monje, C.A., Vinagre, B.M., Chen, Y.Q., Feliu, V., Lanusse, P., and Sabatier, J. (2005). Optimal tunings for fractional $PI^{\lambda}D^{\mu}$–controllers. *Fractional Differentiation and its Applications*. Le Mehauté, A., Tenreiro Machado, J.A., Trigeassou, J.C., and Sabatier, J. (eds.) Ubooks Verlag, Augsburg, pp. 675-686.

Podlubny, I. (1999). Fractional-order systems and $PI^{\lambda}D^{\mu}$-controllers. *IEEE Trans. Autom. Control*. Vol. 44 (1), pp. 208-214.

Shinskey, F.G. (1996). *Process Control Systems. Application, design, and tuning*. 4th edition. McGraw-Hill

Vinagre, B.M., Podlubny, I., Dorcak, L. and Feliu, V. (2000). On fractional PID controllers: A frequency domain approach. *Proceedings of IFAC workshop on Digital Control,* pp. 51-56. Terrassa, Spain.

Ziegler, J.G. and Nichols N.B. (1942). Optimum settings for automatic controllers. *Transactions ASME,* Vol. 64, pp. 759-768.

On A New Approach for Self-optimizing Control Structure Design

S. Heldt *

* *Linde AG, Linde Engineering Division, Dr.-Carl-von-Linde-Str. 6-14 82049 Pullach (Tel: +498974453536; e-mail: steffen.heldt@linde-le.com).*

Abstract
In this paper, a new method for the identification of self-optimizing control structure designs (CSDs) based on generalized singular value decomposition (GSVD) is proposed. The method is primarily dedicated to find optimal CSDs where all controlled variables (CVs) are represented by a common set of linear combinations of process variables (PVs). It is shown that the implementation of the GSVD into iterative solution approaches is beneficial in order to find CSDs where an individual PV subset is mapped to each CV. The developments will be tested on a simple process.

Keywords: Control system design; Linear control systems; Controlled variables; Optimal control; Self-optimizing control

1. INTRODUCTION

With more than 4000 completed plant projects, the Engineering Division of the Linde AG ranks among the leading international plant contractors, with focus on the key market segments olefin plants, natural gas plants, air separation plants, as well as hydrogen and synthesis gas plants. This paper relates to new process developments for liquefied natural gas plants whose steady state and dynamic behavior are currently under investigation in order to provide design guidelines and to ensure reliable and economic operation. In the context of this work, new methods for the identification of regulatory control structure designs (CSDs) have been developed. They will be presented in this paper.

Steady state process optimization by regulation was first motivated by Morari et al. [1980]. They articulated the idea that a constant set point policy will lead to optimal operation if the underlying control structure is properly designed. Skogestad and Postlethwaite [1996, p. 428-433] extended this idea and gave an approximate criterion for finding CSDs with self-optimizing abilities, the so-called minimum singular value (MSV) rule. Assuming a linear process model and a quadratic cost function, an exact local criterion for the worst-case loss of CSDs was developed by Halvorsen et al. [2003]. Based on the resulting worst-case loss criterion, a multivariate non-convex problem subject to structural constraints needs to be solved in order to obtain a self-optimizing CSD. These structural constraints refer to limitations on the size of the process variable (PV) subset and the particular selection of PVs. For instance, it must be decided whether only PV selection or PV combination is taken into account. Several methods have been developed which solve the constrained optimization problem. An optimal CSD subject to PV selection can be found by either screening all possible PV combinations or applying branch and bound (BAB) algorithms in order to avoid time consuming calculations as proposed by Cao and Saha [2005], Kariwala and Skogestad [2006/07/09-13], Cao and Kariwala [2008], Kariwala and Cao [2009]. PV combination methods have been published by Alstad and Skogestad [2007], Alstad et al. [2008], Kariwala [2007] and Kariwala et al. [2008]. They all have in common that the same PV subset is considered for all CVs.

In Section 2, the mathematical framework of self-optimizing control theory will be briefly introduced. In the following sections, two variants of a new PV combination method will be proposed. In Section 3 the focus is on CSDs where a common PV subset is considered for all CVs. Section 4 is dedicated to CSDs where individual PV subsets are mapped to each CV. For illustration, the new developments will be applied to a process example in Section 5. Concluding remarks are given in Section 6.

2. MATHEMATICAL FRAMEWORK

The scheme in Figure 1 represents a general regulatory CSD applied to an arbitrary process plant. Based thereon the exact local method by Halvorsen et al. [2003] will be introduced. The vectors $\boldsymbol{u} \in \mathbb{R}^{n_u}$, $\boldsymbol{d} \in \mathbb{R}^{n_d}$, $\boldsymbol{y} \in \mathbb{R}^{n_y}$ and $\boldsymbol{c} \in \mathbb{R}^{n_u}$, respectively, correspond to the manipulated variables (MVs), disturbance variables (DVs), measured process variables (PVs) and controlled variables (CVs). A constant set point policy is applied. That is, the MVs are adjusted by the controller(s) until, feasibility provided, the CVs equal the set point vector $\boldsymbol{c}_s$. To account for measurement errors, the PVs and CVs are affected by the implementation errors $\boldsymbol{n}^{\boldsymbol{y}} \in \mathbb{R}^{n_y}$ and $\boldsymbol{n}^{\boldsymbol{c}} \in \mathbb{R}^{n_u}$.

Morari et al. [1980], the inventors of self-optimizing control, state that it is desirable "(...) to find a function of PVs which when held constant, leads automatically to the optimal adjustments of the MVs, and with it, the optimal operating conditions." In other words, self-optimizing con-

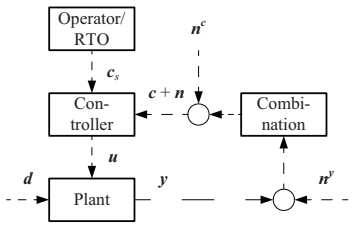


Figure 1. General representation of regulatory CSDs in chemical plants (after Alstad et al. [2008]).

trol may be achieved by an appropriate mapping of PVs towards CVs, denoted by $\boldsymbol{c} = \mathcal{H}(\boldsymbol{y})$, where $\mathcal{H} \in \mathbb{R}^{n_{\boldsymbol{u}}}$ represents the "combination" block in Figure 1. For deriving the exact local method, Halvorsen et al. [2003] considered a linear map $\boldsymbol{H} = \frac{\partial \boldsymbol{c}}{\partial \boldsymbol{y}^T}$. The cost function of a plant denoted by J are usually affected by both, MVs and DVs. In order to operate the plant optimally (at minimum cost), MVs need to be adjusted subject to variations in DVs. The solution to the problem

$$\boldsymbol{u}_{\text{opt}}(\boldsymbol{d}) = \arg\left(\min_{\boldsymbol{u}} J(\boldsymbol{u}, \boldsymbol{d})\right) \text{ s.t. } g(\boldsymbol{y}, \boldsymbol{u}, \boldsymbol{d}) = 0 \quad (1)$$

gives the best input leading to the lowest achievable cost. Problem (1) will be referred to as feed-forward re-optimization. Here $g \in \mathbb{R}^{n_{\boldsymbol{y}}}$ denotes the steady-state model equations of the plant. The following simplifying assumptions are made.

(1) Nonlinearities of the plant are treated as locally negligible. Then, the steady-state I/O model of the plant can be represented as
$$\boldsymbol{y} = \boldsymbol{G^y}\,\boldsymbol{u} + \boldsymbol{G^y_d}\,\boldsymbol{d}, \quad (2)$$
where $\left[\,\boldsymbol{G^y}\;\;\boldsymbol{G^y_d}\,\right] = -\left(\frac{\partial g}{\partial \boldsymbol{y}^T}\right)^{-1} \frac{\partial g}{\partial\left[\,\boldsymbol{u}^T\;\;\boldsymbol{d}^T\,\right]}$.

(2) The cost function J is locally approximated by a second order Taylor series.

(3) The number of MVs might be reduced as some of them need to be spend in "a priori" controller loops in order to either stabilize the plant or fulfill optimally active constraints. It is assumed that $\boldsymbol{u}$ represents only the remaining MVs available for self-optimizing CSD. The "a priori" controller loops are considered part of the model equations g.

Figure 2 shows exemplarily the operational cost of a process plant versus one DV. The cost of feed-forward re-optimization is indicated by the solid line and gives the lower bound for the cost of feedback control with constant set points. It is thus convenient to define a loss function as

$$L(\boldsymbol{d}) = J(\boldsymbol{u}_{\boldsymbol{H}}(\boldsymbol{d}), \boldsymbol{d}) - J(\boldsymbol{u}_{\text{opt}}(\boldsymbol{d}), \boldsymbol{d}).$$

Here $\boldsymbol{u}_{\boldsymbol{H}}(\boldsymbol{d})$ represents the influence from DVs to MVs for feedback control, easily derived for the linear case. From $\boldsymbol{c} = \boldsymbol{H}\,\boldsymbol{y} \stackrel{!}{=} \boldsymbol{c}_s = \boldsymbol{0}$ and (2) it follows that

$$\boldsymbol{u}_{\boldsymbol{H}} = -\left(\boldsymbol{H}\,\boldsymbol{G^y}\right)^{-1}\,\boldsymbol{H}\,\boldsymbol{G^y_d}\,\boldsymbol{d}.$$

According to Halvorsen et al. [2003] the worst-case loss is given by

$$L_{\text{worst}} = \frac{1}{2}\,\boldsymbol{z}^T\,\boldsymbol{z}, \quad (3)$$

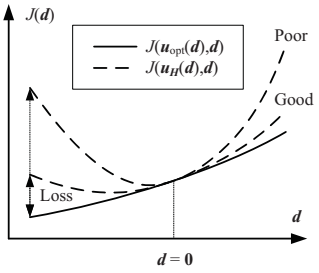


Figure 2. Objective functions for a poor and a good self-optimizing CSD compared with the case of re-optimized MVs (after Skogestad [2000]).

where the loss variables $\boldsymbol{z}$ are given by

$$\boldsymbol{z} = \boldsymbol{M}\,\boldsymbol{f}, \quad (4)$$

for feedback control with

$$\boldsymbol{M} = J_{\boldsymbol{uu}}^{1/2}\,\left(\boldsymbol{H}\,\boldsymbol{G^y}\right)^{-1}\,\boldsymbol{H}\,\tilde{\boldsymbol{F}}$$
$$\tilde{\boldsymbol{F}} = \left[\,-\left(\boldsymbol{G^y}\,J_{\boldsymbol{uu}}^{-1}\,J_{\boldsymbol{ud}} - \boldsymbol{G^y_d}\right)\,\boldsymbol{W_d}\;\boldsymbol{W_{n^y}}\,\right].$$

Here, the matrices $J_{\boldsymbol{uu}}$ and $J_{\boldsymbol{ud}}$ indicate the second derivatives (Hessians) of the cost function J with respect to $\boldsymbol{u}$ and $\boldsymbol{d}$. The disturbance variation $\Delta\boldsymbol{d}$ and the implementation error $\boldsymbol{n_y}$ are commonly represented by the scaled variable $\boldsymbol{f}$, *i.e.*,

$$\begin{bmatrix}\Delta\boldsymbol{d}\\ \boldsymbol{n_y}\end{bmatrix} = \begin{bmatrix}\boldsymbol{W_d} & \boldsymbol{0}\\ \boldsymbol{0} & \boldsymbol{W_{n^y}}\end{bmatrix}\,\boldsymbol{f} \text{ with } \|\boldsymbol{f}\|_2 \le 1,$$

where the matrices $\boldsymbol{W_d}$ and $\boldsymbol{W_{n^y}}$ are diagonal scaling matrices. From observation of (3) and (4) it is evident that a self-optimizing CSD with least worst-case loss may be obtained by solving the problem

$$\boldsymbol{H} = \arg\min_{\boldsymbol{H}} \overline{\sigma}(\boldsymbol{M}), \quad (5)$$

where $\overline{\sigma}$ indicates the largest singular value. In a more recent work, Kariwala et al. [2008] proved that the average loss given by

$$L_{\text{average}} = \frac{1}{6\,(n_{\boldsymbol{y}} + n_{\boldsymbol{d}})}\,\|\boldsymbol{M}\|_{\text{F}}^2 \quad (6)$$

is a better estimate of the loss as the worst-case loss (3) tends to overestimation. Here, $\|.\|_F$ indicates the Frobenius norm also known as the Euclidean norm. Besides, Kariwala et al. [2008] proved that the average loss is super-optimal in the sense that it also minimizes the worst-case loss. According to (6), they suggested solving

$$\boldsymbol{H} = \arg\min_{\boldsymbol{H}} \|\boldsymbol{M}\|_{\text{F}} \quad (7)$$

instead of (5).

The solution of problems (5) and (7) is nontrivial since the matrix $\boldsymbol{M}$ depends in a nonlinear fashion on $\boldsymbol{H}$. Moreover, the problem may be structurally constrained, as indicated in Section 1. *E.g.*, the dimension of the PV subset could be limited or special PVs may be excluded from PV subset etc. Many authors such as Alstad and Skogestad [2007], Alstad et al. [2008], Kariwala [2007] and Kariwala et al. [2008] addressed the problem of finding a global solution to either (5) or (7) with focus on PV combination. All of these

methods are limited to the structural constraint that the same PV subset is used for each CV. Based on the method developed in the next section, iterative solution strategies will be developed which focus on finding CSDs without structural limitations except for rank $(\boldsymbol{H}) = n_{\boldsymbol{u}}$.

3. THE GSVD METHOD

In this section a new solution method is presented for the worst-case and average loss problem, (5) and (7), subject to a common PV subset for all CVs. It will be referred to as the GSVD method. In a first step, (4) is restated as

$$\boldsymbol{z}^T \left(\boldsymbol{G}_{\boldsymbol{z}}^{\boldsymbol{y}}\right)^T \boldsymbol{H}^T = \boldsymbol{f}^T \tilde{\boldsymbol{F}}^T \boldsymbol{H}^T, \tag{8}$$

where

$$\boldsymbol{G}_{\boldsymbol{z}}^{\boldsymbol{y}} = \boldsymbol{G}^{\boldsymbol{y}} J_{\boldsymbol{uu}}^{-1/2}.$$

Suppose, that the rank condition

$$\operatorname{rank}\left(\begin{bmatrix} \left(\boldsymbol{G}_{\boldsymbol{z}}^{\boldsymbol{y}}\right)^T \\ \tilde{\boldsymbol{F}}^T \end{bmatrix}\right) = n_{\boldsymbol{y}} \tag{9}$$

is satisfied which will be generally the case if the condition $n_{\boldsymbol{u}} + n_{\boldsymbol{f}} \geq n_{\boldsymbol{y}}$ holds (the case $n_{\boldsymbol{u}} + n_{\boldsymbol{f}} < n_{\boldsymbol{y}}$ is discussed below in Remark 4). Then, according to Hogben [2007, p. 15.12f], the generalized singular value decomposition (GSVD) of the matrix pair $\left\{\left(\boldsymbol{G}_{\boldsymbol{z}}^{\boldsymbol{y}}\right)^T, \tilde{\boldsymbol{F}}^T\right\}$ exists and (8) can be written as

$$\boldsymbol{z}^T \boldsymbol{U} \boldsymbol{\Sigma} \boldsymbol{V}^T \boldsymbol{H}^T = \boldsymbol{f}^T \tilde{\boldsymbol{U}} \tilde{\boldsymbol{\Sigma}} \boldsymbol{V}^T \boldsymbol{H}^T, \tag{10}$$

where the decomposed matrices have the following properties.

The matrices $\boldsymbol{U} \in \mathbb{R}^{n_{\boldsymbol{u}} \times n_{\boldsymbol{u}}}$ and $\tilde{\boldsymbol{U}} \in \mathbb{R}^{n_{\boldsymbol{f}} \times n_{\boldsymbol{f}}}$ are unitary, *i.e.*, $\boldsymbol{U}^T \boldsymbol{U} = \boldsymbol{I}_{n_{\boldsymbol{u}}}$ and $\tilde{\boldsymbol{U}}^T \tilde{\boldsymbol{U}} = \boldsymbol{I}_{n_{\boldsymbol{f}}}$. The matrix $\boldsymbol{V} \in \mathbb{R}^{n_{\boldsymbol{y}} \times n_{\boldsymbol{y}}}$ is regular. The matrix $\boldsymbol{\Sigma} \in \mathbb{R}^{n_{\boldsymbol{u}} \times n_{\boldsymbol{y}}}$ is tailing diagonal, with $\boldsymbol{\Sigma}^T \boldsymbol{\Sigma} = \operatorname{diag}\left(\alpha_1^2, \alpha_2^2, ..., \alpha_{n_{\boldsymbol{y}}}^2\right)$ and $0 \leq \alpha_i \leq \alpha_{i+1} \leq 1$. The matrix $\tilde{\boldsymbol{\Sigma}} \in \mathbb{R}^{n_{\boldsymbol{f}} \times n_{\boldsymbol{y}}}$ is leading diagonal, with $\tilde{\boldsymbol{\Sigma}}^T \tilde{\boldsymbol{\Sigma}} = \operatorname{diag}\left(\beta_1, \beta_2, ..., \beta_{n_{\boldsymbol{y}}}\right)$ and $1 \geq \beta_i \geq \beta_{i+1} \geq 0$. Note that the number of $r = \max\left(0, n_{\boldsymbol{y}} - n_{\boldsymbol{u}}\right)$ leading α_i and β_i are 0 and 1, respectively, and that the number of $s = \max\left(0, n_{\boldsymbol{y}} - n_{\boldsymbol{f}}\right)$ tailing α_i and β_i are 1 and 0, respectively. For more information on GSVD and on how the resulting matrices can be computed, the reader is referred to standard linear algebra textbooks, *e.g.*, Golub and VanLoan [1996, pp. 465-467].

Theorem 1. If $n_{\boldsymbol{y}} \geq n_{\boldsymbol{u}}$, the minimum worst-case and average loss are given by

$$L_{\text{worst}} = \frac{1}{2}\left(\frac{\beta_{r+1}}{\alpha_{r+1}}\right)^2 \tag{11}$$

and

$$L_{\text{average}} = \frac{1}{6\left(n_{\boldsymbol{y}} + n_{\boldsymbol{d}}\right)} \sum_{i=r+1}^{n_{\boldsymbol{y}}} \left(\frac{\beta_i}{\alpha_i}\right)^2, \tag{12}$$

respectively. They may be obtained by selecting

$$\boldsymbol{H} = \boldsymbol{M}_{\boldsymbol{n}} \begin{bmatrix} \boldsymbol{p}_{r+1} & \cdots & \boldsymbol{p}_{n_{\boldsymbol{y}}} \end{bmatrix}^T,$$

where $\boldsymbol{p}_i$ is the i^{th} column of $\boldsymbol{P} = \boldsymbol{V}^{-T}$ and $\boldsymbol{M}_{\boldsymbol{n}} \in \mathbb{R}^{n_{\boldsymbol{u}} \times n_{\boldsymbol{u}}}$ is an arbitrary regular matrix.

Proof. By selecting $\boldsymbol{H} = \boldsymbol{M}_{\boldsymbol{n}} \begin{bmatrix} \boldsymbol{p}_{r+1} & \cdots & \boldsymbol{p}_{n_{\boldsymbol{y}}} \end{bmatrix}^T$ it follows that

$$\boldsymbol{V}^T \boldsymbol{H}^T = \boldsymbol{V}^T \begin{bmatrix} \boldsymbol{p}_{r+1} & \cdots & \boldsymbol{p}_{n_{\boldsymbol{y}}} \end{bmatrix} \boldsymbol{M}_{\boldsymbol{n}}^T = \begin{bmatrix} \boldsymbol{0}_{r \times n_{\boldsymbol{u}}} \\ \boldsymbol{I}_{n_{\boldsymbol{u}}} \end{bmatrix} \boldsymbol{M}_{\boldsymbol{n}}^T. \tag{13}$$

and

$$\boldsymbol{\Sigma} \boldsymbol{V}^T \boldsymbol{H}^T = \operatorname{diag}\left(\alpha_{r+1}, ..., \alpha_{n_{\boldsymbol{y}}}\right) \boldsymbol{M}_{\boldsymbol{n}}^T$$

$$\tilde{\boldsymbol{\Sigma}} \boldsymbol{V}^T \boldsymbol{H}^T = \begin{bmatrix} \boldsymbol{0}_{r \times n_{\boldsymbol{u}}} \\ \operatorname{diag}\left(\beta_{r+1}, ..., \beta_{n_{\boldsymbol{y}}}\right) \\ \boldsymbol{0}_{\bar{s} \times n_{\boldsymbol{u}}} \end{bmatrix} \boldsymbol{M}_{\boldsymbol{n}}^T,$$

where $\bar{s} = \max\left(0, n_{\boldsymbol{f}} - n_{\boldsymbol{y}}\right)$. Inserting these results into (10) yields

$$\boldsymbol{z}^T = \boldsymbol{f}^T \underbrace{\tilde{\boldsymbol{U}} \begin{bmatrix} \boldsymbol{0}_{r \times n_{\boldsymbol{u}}} \\ \operatorname{diag}\left(\sigma_{r+1}, ..., \sigma_{n_{\boldsymbol{y}}}\right) \\ \boldsymbol{0}_{\bar{s} \times n_{\boldsymbol{u}}} \end{bmatrix}}_{=\boldsymbol{M}^T} \boldsymbol{U}^T,$$

where $\sigma_i = \beta_i / \alpha_i$ indicate the i^{th} largest generalized singular value of the matrix pair $\left\{\left(\boldsymbol{G}_{\boldsymbol{z}}^{\boldsymbol{y}}\right)^T, \tilde{\boldsymbol{F}}^T\right\}$. Note that the maximum singular value and the Frobenius norm of a matrix are invariant to unitary transformations thereof. Thus, the worst-case loss (3) and the average loss (6) depend only on the selected generalized singular values and the derivation of (11) and (12) is trivial. As the minimum generalized singular values were selected, both, the worst-case loss and the average loss are minimal.

Remark 2. The GSVD method is written in terms of the complete PV set $\mathcal{Y}$. Note that it work as well for a selected PV subset $\mathcal{Y}_{\text{c}} \subseteq \mathcal{Y}$. Then, in all formulas stated above the respective rows in $\boldsymbol{G}_{\boldsymbol{z}}^{\boldsymbol{y}}$ and $\tilde{\boldsymbol{F}}$ must be extracted and $n_{\boldsymbol{y}}$ must be substituted by $n_{\mathcal{Y}_{\text{c}}}$.

Remark 3. For perfect disturbance rejection, *i.e.*, $\boldsymbol{M} = \boldsymbol{0}$, it is required that at least $n_{\boldsymbol{u}}$ tailing β_i are 0. As indicated above, the number of $s = \max\left(0, n_{\boldsymbol{y}} - n_{\boldsymbol{f}}\right)$ tailing β_i are in fact 0. Thus, perfect disturbance rejection occurs if the inequality

$$s \geq n_{\boldsymbol{u}}$$

is satisfied. The necessary condition therefore is $s > 0$, *i.e.*, $n_{\boldsymbol{f}} < n_{\boldsymbol{y}}$, which can only be satisfied if the implementation error is disregarded, *i.e.*, $\boldsymbol{W}_{\boldsymbol{n^y}} = \emptyset$ and $n_{\boldsymbol{f}} = n_{\boldsymbol{d}}$. The sufficient condition for perfect disturbance rejection is then $n_{\boldsymbol{y}} \geq n_{\boldsymbol{u}} + n_{\boldsymbol{d}}$. This is in agreement with the combination methods from Alstad and Skogestad [2007], Alstad et al. [2008] and Kariwala [2007], Kariwala et al. [2008].

Remark 4. If $n_{\boldsymbol{u}} + n_{\boldsymbol{f}} < n_{\boldsymbol{y}}$, the rank condition (9) is violated and the GSVD as stated above cannot be performed. Note that this is only a formal issue and will not be treated here for the sake of brevity.

Remark 5. The GSVD method is related to the method by Kariwala et al. [2008] and the "constrained average loss minimization" method by Alstad et al. [2008]. It can be shown that all three methods minimize the average loss subject to the same structural constraint and thus provide the same results. However, it will be omitted here for the sake of brevity.

4. BEYOND COMMON PV SUBSETS

For better legibility of this section, definitions for CSDs will be introduced.

Definition 6. A CSD is said to be *column-structured*, if all CVs are linear combinations of the same PV subset $\mathcal{Y}_\mathrm{c}$ of size $n_{\mathcal{Y}_\mathrm{c}} = \dim(\mathcal{Y}_\mathrm{c})$. A *common-sized* CSD refers to a CSD in which the i^{th} CV is a linear combination of an individual PV subset $\mathcal{Y}_i$ with the constraint that all PV subsets have the same set size $n_\mathrm{s} = \dim(\mathcal{Y}_i)\,\forall i \in \{1,\dots,n_{\boldsymbol{u}}\}$. In a more general *loosely-structured* CSD, the i^{th} CV is a linear combination of an individual PV subset $\mathcal{Y}_i$ with individual set size $n_{\mathcal{Y}_i} = \dim(\mathcal{Y}_i)$.

Theorem 7. Let $\boldsymbol{H}_\mathrm{c}$ represent a column-structured CSD of size $n_{\mathcal{Y}_\mathrm{c}}$ with finite worst-case/average loss, *i.e.*, $\operatorname{rank}(\boldsymbol{H}_\mathrm{c}) = n_{\boldsymbol{u}}$. Then, for every $\boldsymbol{H}_\mathrm{c}$ there exists a common-sized CSD $\boldsymbol{H}_\mathrm{s}$ with a PV subset size of $n_\mathrm{s} = n_{\mathcal{Y}_\mathrm{c}} - (n_{\boldsymbol{u}} - 1)$ and the same worst-case/average loss as $\boldsymbol{H}_\mathrm{c}$. The proof will be omitted due to the lack of space.

Corollary 8. Let $\boldsymbol{H}_\mathrm{c}$ be a column-structured CSD with PV subset size $n_\mathrm{c} = n_{\boldsymbol{u}}$ and finite loss. Then, the worst-case/average loss of $\boldsymbol{H}_\mathrm{c}$ is independent of the coefficients in $\boldsymbol{H}_\mathrm{c}$. Rather, the worst-case/average loss of $\boldsymbol{H}_\mathrm{c}$ depends only on the selection of the PV subset $\mathcal{Y}_\mathrm{c}$. The proof will be omitted due to the lack of space.

Some advantages of common-sized and loosely-structured CSDs over column-structured CSDs are pointed out below.

(1) A smaller PV subset size is, on the one hand, favorable due to better practical acceptance but, on the other hand, usually accompanied by a larger worst-case/average loss. From Theorem 7 it can be concluded that for $n_{\boldsymbol{u}} > 1$ a reduction in PV subset size without affecting the worst-case/average loss can be achieved if, instead of a column-structured CSD, a common-sized CSD is taken into account. In particular, the PV subset size reduction with invariant worst-case/average loss can be as large as $n_{\boldsymbol{u}} - 1$ PVs.
(2) By implication of the first argument, it is evident that a smaller worst-case/average loss can be achieved if, instead of a column-structured CSD, a common-sized CSD with equal PV subset size is taken into account.
(3) For $n_{\mathcal{Y}_\mathrm{c}} = n_{\boldsymbol{u}}$ the optimality of column-structured CSDs is only a matter of PV subset selection as pointed out in Corollary 8 presented above.
(4) Column-structured CSDs $\boldsymbol{H}_\mathrm{c}$ fail if $n_{\mathcal{Y}_\mathrm{c}} < n_{\boldsymbol{u}}$ holds. This is due to the fact that $\operatorname{rank}(\boldsymbol{H}_\mathrm{c}) < n_{\boldsymbol{u}}$ which leads to a singular $\boldsymbol{H}_\mathrm{c}\,\boldsymbol{G}^{\boldsymbol{y}}$ and, by observation of (4) and (5), to infinite loss.
(5) Input/output (I/O) selection based on heuristic rules is a common practice. Physical closeness between CVs and MVs is probably the most common rule, in order to achieve good cause and effect between MVs and CVs. If decentralized controllers are used and the MVs are far apart from each other (*e.g.*, in large scale processes), it is desirable to have an individual PV subset for every CV as in common-sized and loosely-structured CSDs.
(6) PV combinations including different measurement units have poor practical acceptance. If the structural constraint was imposed on the prospective CSD that only PVs of the same type can be selected for each CV, then, in the case of a column-structured CSD with $n_{\boldsymbol{u}} > 1$, one would be forced to omit information of all PVs not part of the selected unit group. In common-sized and loosely-structured CSDs, for each CV another PV subset can be selected which allows to use information of more than only one unit group.

For loosely-structured CSDs, no explicit expression for $\boldsymbol{H}$ can be derived by the solution to problems (5) and (7). Thus, iterative solution methods need to be applied. In the following, a framework for advanced iterative methods will be presented.

In order to take loosely-structured CSDs into account, (4) is restated as

$$\boldsymbol{z}^T \sum_{i=1}^{n_{\boldsymbol{u}}} (\boldsymbol{G}_{\boldsymbol{z}}^{\boldsymbol{y}})^T\, \boldsymbol{h}_i\, \boldsymbol{e}_i^T = \boldsymbol{f}^T \sum_{i=1}^{n_{\boldsymbol{u}}} \tilde{\boldsymbol{F}}^T\, \boldsymbol{h}_i\, \boldsymbol{e}_i^T, \tag{14}$$

where $\boldsymbol{h}_i^T \in \mathbb{R}^{n_{\boldsymbol{y}}}$ is the i^{th} row vector of $\boldsymbol{H}$, *i.e.*, $\boldsymbol{H}^T = [\,\boldsymbol{h}_1\ \dots\ \boldsymbol{h}_{n_{\boldsymbol{u}}}\,]$, and $\boldsymbol{e}_i \in \mathbb{R}^{n_{\boldsymbol{u}}}$ is the i^{th} standard basis vector. The vector $\boldsymbol{h}_i$ represents the map from the PVs of the subset $\mathcal{Y}_i$ towards the i^{th} CV, hence $h_{ij} = 0 \forall j \notin \mathcal{Y}_i$. It is thus convenient to write (14) as

$$\boldsymbol{z}^T \sum_{i=1}^{n_{\boldsymbol{u}}} (\boldsymbol{G}_{\boldsymbol{z}}^{\boldsymbol{y}})_{\mathcal{Y}_i}^T\, \boldsymbol{h}_{i\mathcal{Y}_i}\, \boldsymbol{e}_i^T = \boldsymbol{f}^T \sum_{i=1}^{n_{\boldsymbol{u}}} \tilde{\boldsymbol{F}}_{\mathcal{Y}_i}^T\, \boldsymbol{h}_{i\mathcal{Y}_i}\, \boldsymbol{e}_i^T, \tag{15}$$

where the subscript $\mathcal{Y}_i$ denotes that those columns/elements of $(\boldsymbol{G}_{\boldsymbol{z}}^{\boldsymbol{y}})^T$, $\tilde{\boldsymbol{F}}^T$ and $\boldsymbol{h}_i$ are selected whose index is part of $\mathcal{Y}_i$.

By performing the GSVD of the corresponding matrix pairs $\left\{(\boldsymbol{G}_{\boldsymbol{z}}^{\boldsymbol{y}})_{\mathcal{Y}_i}^T, \tilde{\boldsymbol{F}}_{\mathcal{Y}_i}^T\right\}$, (15) can be written as

$$\boldsymbol{z}^T \underbrace{\sum_{i=1}^{n_{\boldsymbol{u}}} \boldsymbol{U}_i\, \boldsymbol{I}_{n_{\boldsymbol{u}} \times n_{\mathcal{Y}_i}}\, \tilde{\boldsymbol{h}}_i\, \boldsymbol{e}_i^T}_{=\boldsymbol{X}} = \boldsymbol{f}^T \sum_{i=1}^{n_{\boldsymbol{u}}} \tilde{\boldsymbol{U}}_i\, \boldsymbol{\mathcal{S}}_i\, \tilde{\boldsymbol{h}}_i\, \boldsymbol{e}_i^T, \tag{16}$$

where $\boldsymbol{\mathcal{S}}_i = \left[\, \boldsymbol{0}_{n_{\mathcal{Y}_i} \times r_i}\ \operatorname{diag}\left(\sigma_{i1}, ..., \sigma_{in_{\mathcal{Y}_i}}\right)\ \boldsymbol{0}_{n_{\mathcal{Y}_i} \times \bar{s}_i}\,\right]^T$, $r_i = \max(0, n_{\mathcal{Y}_i} - n_{\boldsymbol{u}})$, $\bar{s}_i = \max(0, n_{\boldsymbol{f}} - n_{\mathcal{Y}_i})$ and $\sigma_{ij} = \begin{cases} \alpha_{ij}/\beta_{ij} & \text{if } \beta_{ij} \neq 0 \\ 1 & \text{otherwise} \end{cases}$; $\boldsymbol{U}_i \in \mathbb{R}^{n_{\boldsymbol{u}} \times n_{\boldsymbol{u}}}$ and $\tilde{\boldsymbol{U}}_i \in \mathbb{R}^{n_{\boldsymbol{f}} \times n_{\boldsymbol{f}}}$ are unitary; $\boldsymbol{I}_{n_{\boldsymbol{u}} \times n_{\mathcal{Y}_i}}$ is the $n_{\boldsymbol{u}} \times n_{\mathcal{Y}_i}$ tailing diagonal identity matrix; and $\tilde{\boldsymbol{h}}_i = \operatorname{diag}\left(\beta_{i(r_i+1)}, ..., \beta_{in_{\mathcal{Y}_i}}\right) \boldsymbol{V}_i^T\, \boldsymbol{h}_{i\mathcal{Y}_i}$, with $\boldsymbol{V}_i \in \mathbb{R}^{n_{\mathcal{Y}_i} \times n_{\mathcal{Y}_i}}$ if $n_{\boldsymbol{u}} + n_{\boldsymbol{f}} \geq n_{\mathcal{Y}_i}$. The decomposed formulation (16) has several advantages over (10) as pointed out below.

(1) From observation of (16) it can be seen that the first $r_i = \max(0, n_{\mathcal{Y}_i} - n_{\boldsymbol{u}})$ elements in $\tilde{\boldsymbol{h}}_i$ do not contribute to $\boldsymbol{X}$. It is generally close to optimal to set them to zero as the corresponding columns in $\tilde{\boldsymbol{U}}_i\, \boldsymbol{\mathcal{S}}_i$ vanish. Thus, if any $r_i > 0$ then one can reduced the variable space from $\sum_{i=1}^{n_{\boldsymbol{u}}} n_{\mathcal{Y}_i}$ to $\sum_{i=1}^{n_{\boldsymbol{u}}} n_{\mathcal{Y}_i} - r_i$. The maximum dimension of the reduced space is $n_{\boldsymbol{u}} \times n_{\boldsymbol{u}}$. (16). This approach will be referred to as the reduced space (RC) method. The starting values for the RC method will be $\tilde{\boldsymbol{h}}_i = \boldsymbol{e}_{n_{\mathcal{Y}_i}}$ which corresponds to the selection of the smallest generalized singular values.
(2) From (16), it can be shown that a suboptimal solution to (7) can be obtained by setting the first r_i elements in $\tilde{\boldsymbol{h}}_i$ are zero and solving the substitute problem

$$\boldsymbol{X} = \arg\min_{\boldsymbol{X}} \sum_{i=1}^{n_{\boldsymbol{u}}} \boldsymbol{X}_i^T\, \tilde{\boldsymbol{\mathcal{S}}}_i^T\, \tilde{\boldsymbol{\mathcal{S}}}_i\, \boldsymbol{X}_i \text{ s.t. } \boldsymbol{X}^T\, \boldsymbol{X} = \boldsymbol{I}, \tag{17}$$

where $\tilde{\boldsymbol{\mathcal{S}}}_i = \boldsymbol{\mathcal{S}}_i\, \boldsymbol{I}_{n_{\boldsymbol{u}} \times n_{\mathcal{Y}_i}}^T\, \boldsymbol{U}_i^T$ and $\boldsymbol{X}_i$ is the i^{th} column of $\boldsymbol{X}$. Problem (17) is still nonconvex but has the

advantage that an efficient steepest descend method can be developed. Due to the lack of space, the method cannot be outlined here but will be an issue of a subsequent publication. It will be referred to as the unitary matrix constraint (UMC) method. The starting value for the iterative solution of $\boldsymbol{X}$ will be the identity matrix.

(3) From (16) a lower bound for the minimum worst-case/average loss can be derived. This is particularly helpful in reducing computational expense as described below.

The large number of alternative control structures can be reduced by excluding candidate CVs which cannot lead to an optimal solution. This strategy is known as the BAB principle. BAB algorithms have been formerly applied to CSD problems by several authors such as Cao and Saha [2005], Kariwala and Skogestad [2006/07/09-13], Cao and Kariwala [2008] and Kariwala and Cao [2009]. Lower bounds on the minimal worst-case/average loss are helpful for discriminating candidate CVs. From the conclusion that the lower bounds corresponds to the ideal case that $\tilde{\boldsymbol{h}}_i = \boldsymbol{e}_{n_{\mathcal{Y}_i}}$, $\boldsymbol{U}_{i\mathcal{Y}_i} \perp \boldsymbol{U}_{j\mathcal{Y}_j}$ and $\tilde{\boldsymbol{U}}_{i\mathcal{Y}_i} \perp \tilde{\boldsymbol{U}}_{j\mathcal{Y}_j} \forall i \neq j$ it follows from observation of (16) that

$$\boldsymbol{M} = \sum_{i=1}^{n_{\boldsymbol{u}}} \sigma_{in_{\mathcal{Y}_i}} \boldsymbol{e}_i \boldsymbol{e}_i^T.$$

This yields the inequalities

$$L_{\text{worst}} \geq \frac{1}{2} \max_i \left(\sigma^2_{in_{\mathcal{Y}_i}} \right) \tag{18}$$

$$L_{\text{average}} \geq \frac{1}{6\ (n_{\mathcal{Y}_\cap} + n_{\boldsymbol{d}})} \sum_{i=1}^{n_{\boldsymbol{u}}} \sigma^2_{in_{\mathcal{Y}_i}}. \tag{19}$$

Note that $n_{\mathcal{Y}_\cap}$ indicates the size of the merged PV subset $\mathcal{Y}_1 \cap \cdots \cap \mathcal{Y}_{n_{\boldsymbol{u}}}$. It is important to state that (18) and (19) also hold for incomplete set of candidate CVs, *i.e.*, the lower bound of one candidate CV is also the lower bound of all possible control structures which include this CV. If an upper bound for the worts-case/average loss of all alternatives L_{ub} is known, the evaluation of structures (and substructures) can be omitted which show a lower bound $L_{\text{lb}} > L_{\text{ub}}$. Unfortunately, the bounds given above are not very tight, so that computational savings are relatively small.

5. EVAPORATOR CASE STUDY

In this section, the proposed CSD methods will be applied to the evaporation process presented in Figure 3. This forced-circulation evaporation was originally treated by Newell and Lee [1989] and has been investigated subsequently by Heath et al. [2000] and Kariwala et al. [2008], among others. The purpose of the process is the concentration of dilute liquor from the feed to the product stream by evaporation and separation of the solvent. The analytic model equations including the cost function and operational constraints can be found in Kariwala et al. [2008]. The process model has three state variables, the level L_2, the composition X_2 and the pressure P_2 with eight degrees of freedom. Table 1 lists the important stream properties, their value at the nominal operating point and

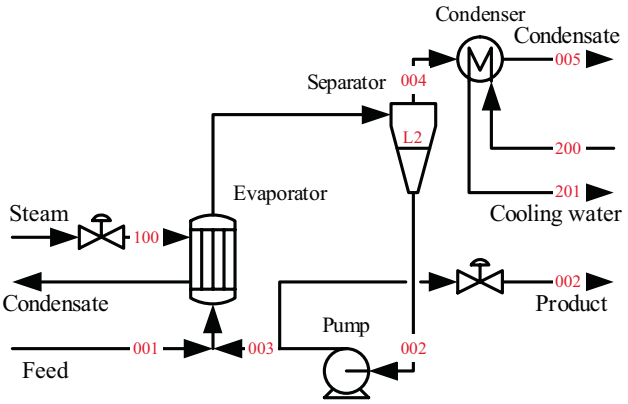


Figure 3. Evaporation process scheme

Var.	Description	Nominal value	Classification
F_1	Feed flow rate	9.469 kg/min	MV, PV (±2 %)
F_2	Product flow rate	1.334 kg/min	MV†, PV (±2 %)
F_3	Circulating flow rate	24.721 kg/min	MV†, PV (±2 %)
F_4	Vapor flow rate	8.135 kg/min	
F_5	Condensate flow rate	8.135 kg/min	PV (±2 %)
X_1	Feed composition	5.00 %	DV (±5 %)
X_2	Product composition	35.50 %	
T_1	Feed temperature	40.0 °C	DV (±20 %)
T_2	Product temperature	88.4 °C	PV (±1 °C)
T_3	Vapor temperature	81.066 °C	PV (±1 °C)
p_2	Operating pressure	51.412 kPa	PV (±2.5 %)
F_{100}	Steam flow rate	9.434 kg/min	PV (±2 %)
T_{100}	Steam temperature	151.52 °C	
p_{100}	Steam pressure	400.0 kPa	MV†
Q_{100}	Heat duty	345.292 kW	
F_{200}	Water flow rate	217.738 kg/min	MV, PV (±2 %)
T_{200}	Water inlet temp.	25.0 °C	DV (±20 %)
T_{201}	Water outlet temp.	45.55 °C	PV (±1 °C)
Q_{200}	Condenser duty	313.21 kW	
J	Operational cost	-582.233 $/h	

Table 1. Key process variables in the evaporation process

their classification into MVs, DVs and PVs. Three out of five MVs indicated by † are used to keep the three PVs L_2, X_2 and P_{100} at their set points. Note that the level in the separator L_2 has no steady-state effect but needs to be controlled for stabilization. The other two controlled PVs need to be kept at their constraints in order to achieve optimality over the given disturbance region. Generality is not lost by this particular selection of the unconstrained MVs. In Table 1, the (embraced) expected variations of the DVs and measurement errors of the PVs are given in % from their nominal value except for temperature measurement errors which are indicated on an absolute scale.

The model equations were implemented in a modeling environment (ME) of the in-house tool OPTISIM® [1], an equation-oriented process simulator. The model has been optimized with respect to the DVs' nominal values given in Table 1 and operational constraints. This led to the operating conditions of the MVs and DVs presented in Table 1. As the ME provides first derivatives by automatic differentiation, the linear I/O gains $G_{\boldsymbol{u}}^{\boldsymbol{y}}$ and $G_{\boldsymbol{u}}^{\boldsymbol{y}}$ at the operating point are directly available. Second derivatives $J_{\boldsymbol{uu}}$ and $J_{\boldsymbol{ud}}$ were estimated by finite difference approx-

[1] OPTISIM® is a registered trademark of the Linde AG. (Burr [1991/4/7-11])

$n_{\mathcal{Y}_c}$	Best PV set	L_{average} (in \$/h)	L_{worst} (in \$/h)
2	F_3, F_{200}	3.8079	56.7126
3	F_2, F_{100}, F_{200}	0.6533	11.6643
4	F_2, T_{201}, F_3, F_{200}	0.4545	9.4516
...			
10	All PVs	0.1941	7.5015

Table 2. Worst-case/average loss of best column-structured CSDs

imation and the use of the NAG routine E04XAF. The numerical results of the I/O gains and the Hessians are in agreement with those of Kariwala et al. [2008]. CSDs for the evaporation process have been identified using the methods presented in Sections 3 and 4. The calculations were conducted in Matlab®R2008b using a Windows XP SP2 desktop with an Intel®Core™Duo CPU E8400 (3.0 Ghz, 3.5 GB RAM).

At first, column-structured CSDs were identified by average loss minimization using the GSVD method. The best control structure was determined by screening over all possibilities satisfying the PV subset size condition imposed. Some results are given in Table 2. They reproduce the results by Kariwala et al. [2008] with an deviation of less than 0.6 %. Both, the minimum worst-case and average loss of the best structure decrease with the PV subset size and approach a lower bound (at $n_{\mathcal{Y}_c} = 10$) asymptotically. According to Corollary 8, the case $n_{\mathcal{Y}_c} = n_{\boldsymbol{u}} = 2$ indicates as well the best PV selection structure.

Next, common-sized CSDs PVs were sought. According to Theorem 7, for each $n_{\mathcal{Y}_c}$-sized column-structured CSD there exists a common-sized CSD of size $n_s = (n_{\mathcal{Y}_c} - n_{\boldsymbol{u}} + 1)$ which can be obtained by a simple linear transformation of the former. Thus, the results in Table 2 indicate also possible common-sized CSDs of PV subset size n_s from one to nine. For instance, Table 3 shows the transformation of the best column-structured CSD with set size three, indicated by $\boldsymbol{H}_{c3}$, into $\boldsymbol{H}_{s2}$ where only combinations of two PVs per CV occur. Despite its small PV subset size, $\boldsymbol{H}_{s2}$ achieves a considerably small average loss. Note that CSDs obtained by this approach are generally not the best among all CSDs satisfying the particular structural constraint of a common-sized set with set size two. In order to find a CSD with lower average loss, the RC method was applied. The best solution found among the $\left(C_{10}^2\right)\left(C_{10}^2 - 1\right)/2 = 990$ alternatives is indicated as $\hat{\boldsymbol{H}}_{s2}$ in Table 3. Due to the BAB algorithm only 103 problems with an average of 0.07 s expense per problem had to be solved. The total computation time was 8.3 s. Using the UMC method, the computational efficiency could be reduced to 6 ms expense per problem leading to a total computation time of 3.3 s at 144 iterative problem solutions. The solution $\tilde{\boldsymbol{H}}_{s2}$ showed a larger average loss than $\hat{\boldsymbol{H}}_{s2}$ structurally identical though.

Suppose that due to cost issues, only one flow meter can be afforded. Since temperature and pressure indicators are rather cheap, their numbers are not limited by cost considerations. In this situation, the task is to find the two best CVs out of $C_6^1 + 2$ candidates, *i.e.*, one out of six flows, one pressure and one temperature set. The best CSD indicated as $\boldsymbol{H}_{s1F}$ in Table 3 was found by applying

CSD	L_{average} (in \$/h)
$\boldsymbol{H}_{c3} = \begin{bmatrix} -0.99 & 0.15 & 0.00 \\ -0.99 & -0.12 & 0.01 \end{bmatrix} \begin{bmatrix} F_2 & F_{100} & F_{200} \end{bmatrix}^T$	0.6533
$\boldsymbol{H}_{s2} = \begin{bmatrix} -6.27 & 1.0 & \\ -143.08 & & 1.0 \end{bmatrix} \begin{bmatrix} F_2 & F_{100} & F_{200} \end{bmatrix}^T$	0.6533
$\hat{\boldsymbol{H}}_{s2} = \begin{bmatrix} -6.27 & 1.0 & & \\ & & 1.0 & -23.30 \end{bmatrix} \begin{bmatrix} F_2 & F_{100} & F_{200} & F_1 \end{bmatrix}^T$	0.5673
$\tilde{\boldsymbol{H}}_{s2} = \begin{bmatrix} -6.22 & 1.0 & & \\ & & 1.0 & -13.34 \end{bmatrix} \begin{bmatrix} F_2 & F_{100} & F_{200} & F_1 \end{bmatrix}^T$	0.6682
$\boldsymbol{H}_{s1F} = \begin{bmatrix} 1.0 & & & \\ & 0.36 & 0.33 & 0.87 \end{bmatrix} \begin{bmatrix} F_3 & T_2 & T_3 & T_{201} \end{bmatrix}^T$	2.9704
$\boldsymbol{H}_{c3T} = \begin{bmatrix} 0.59 & 0.53 & -0.61 \\ 0.02 & 0.01 & 1.0 \end{bmatrix} \begin{bmatrix} T_2 & T_3 & T_{201} \end{bmatrix}^T$	3.6573

Table 3. CSD results

the RC method. It shows slightly better average loss than $\boldsymbol{H}_{c3T}$ which is the best column-structured CSD where all temperatures are used.

6. CONCLUSION

In this paper new insights into the identification problem of self-optimizing CSDs were given. The GSVD method was proposed which allows finding CVs, altogether linear combinations of a common PV subset. It minimizes the average loss super-optimal to the worst-case loss by taking expected disturbances and measurement errors into account. The GSVD method can be beneficially implemented into iterative solution approaches in order to find loosely-structured CSDs where for each CV an individual PV subsets is taken into account. The new methods were successfully applied to an evaporation process. It could be shown that loosely structured CSDs are favorable in terms of flexibility, practical acceptance and economic considerations.

REFERENCES

E04XAF. In *NAG Fortran library manual: Mark 21*. 2006. ISBN 1852062045.

V. Alstad and S. Skogestad. *Ind. eng. chem. res.*, 46(3):846–853, 2007.

V. Alstad, S. Skogestad, and S. E. Hori. *J. proc. cont.*, 2008.

S. P. Burr. AIChE spr. nat. meet., Houston, Texas, 1991/4/7-11.

Y. Cao and V. Kariwala. *Comp. chem. eng.*, 32(10):2306–2319, 2008.

Y. Cao and P. Saha. *Chem. eng. sci.*, 60(6):1555–1564, 2005.

H. G. Golub and F. C. VanLoan. Johns Hopkins Univ. Press, Baltimore, Maryland, 3rd ed. edition, 1996. ISBN 0801854148.

J. I. Halvorsen, S. Skogestad, C. J. Marud, and V. Alstad. *Ind. eng. chem. res.*, 42:3273–3284, 2003.

A. J. Heath, K. I. Kookos, and D. J. Perkins. *AIChE*, 46(10):1998–2016, 2000.

L. Hogben. Chapman & Hall/CRC, Boca Raton, Florida, 2007. ISBN 1584885106.

V Kariwala. *Ind. eng. chem. res.*, 46(46):3629–3634, 2007.

V. Kariwala and Y. Cao. *Comp. chem. eng.*, in print, 2009.

V. Kariwala and S. Skogestad. PSE/ESCAPE, Garmisch-Partenkirchen, Germany, 2006/07/09-13.

V. Kariwala, Y. Cao, and S. Janardhanan. *Ind. eng. chem. res.*, 47 (4):1150–1158, 2008.

M. Morari, Y. Arkun, and G. Stephanopoulos. *AIChE*, 26(2):220–232, 1980.

B. R. Newell and L. P. Lee. Prentice-Hall, New York, NY, 1989. ISBN 0130409405.

S. Skogestad. *J. proc. cont.*, 10(5):487–507, 2000.

S. Skogestad and I. Postlethwaite. Wiley, Chichester, UK, 1996. ISBN 0471942774.

An Online Algorithm for Robust Distributed Model Predictive Control

Walid Al-Gherwi. Hector Budman. Ali Elkamel

Chemical Engineering Department, University of Waterloo, Waterloo, Canada

Abstract: Distributed Model Predictive Control (DMPC) has received significant attention in the literature. However, the robustness of DMPC with respect to model errors has not been explicitly addressed. In this paper, an online algorithm that deals explicitly with model errors for DMPC is proposed. The algorithm requires decomposing the entire system into *N* subsystems and solving *N* convex optimization problems to minimize an upper bound on a robust performance objective by using a time-varying state-feedback controller for each subsystem. Simulations on two typical examples were considered to illustrate the application of the proposed method.

Keywords: Distributed Model Predictive Control; Robust Control.

1. INTRODUCTION

Distributed model predictive control (DMPC) has received significant attention in the literature in recent years. The key potential advantages of DMPC are: i) it can provide better performance than fully decentralized control especially when the interactions ignored in the latter are strong, and ii) it can maintain the flexibility with respect to equipment failure and partial plant shutdowns that may jeopardize the successful operation of centralized MPC. The basic idea of DMPC is to partition the total system of states and controlled and manipulated variables into smaller subsystems and to assign an MPC controller to each subsystem. The design of all the reported DMPC strategies is composed of three parts: (1) Modeling; each controller has access to a local dynamic model of the corresponding subsystem along with an interaction dynamic model that represents the influence of the other subsystems. These models can be obtained by directly decomposing a centralized model of the process (Rawlings and Stewart 2008). (2) Optimization; each MPC solves a local optimization problem. Some reported strategies use modified objective functions that take into account the goals of other controllers to achieve full coordination (Venkat 2006; Zhang and Li 2007) whereas some others use strict local objectives (Li *et al.* 2005), e.g. a Nash-equilibrium objective. (3) Communication; at every control time interval all the controllers exchange the measurements of their local states that are used for subsequent local optimization. These 3 steps are executed at each time interval in an iterative manner until convergence among the controllers is reached. Venkat (2006) showed that increasing the iterations allows the DMPC strategy to reach the optimal centralized solution and the termination at any intermediate iteration maintains system-wide feasibility. Zhang and Li (2007) analyzed the optimality of the iterative DMPC scheme and derived closed-form solution for an unconstrained DMPC and showed that it is identical to centralized MPC solution. The common feature of the reported strategies is that they employ a nominal model of the plant and rely on feedback to account for plant-model mismatch. However, plant–model mismatch may have a significant impact on stability and performance. Thus, the robustness of DMPC to model errors has been identified as a key factor for a successful application of DMPC (Rawlings and Stewart 2008). Kothare *et al.* (1996) proposed a methodology for robust centralized constrained MPC design that maintains robust stability and minimizes a bound on performance in the presence of model errors. The problem is formulated as a convex optimization problem with linear matrix inequalities LMI that is solved efficiently using available algorithms (Boyd *et al.* 1994) and can be used for on-line implementations. This method has been recognized as a good potential candidate for use in process industry to handle the issue of plant-model mismatch (Qin and Badgwell 2003).

The aim of this paper is to present a methodology for Robust DMPC (RDMPC) that explicitly deals with model errors. An LMI-based predictive control formulation (Kothare *et al.* 1996) has been modified to design an on-line iterative algorithm for RDMPC. Issues of robust stability and convergence are analyzed and discussed. Two case studies are used to illustrate the algorithm: a distillation column example (Venkat 2006) when "bad" input-output pairings are chosen and a high-purity column example (Skogestad and Morari, 1988) with high condition number.

2. Definitions and Methodology

2.1 Models

In this work, it is assumed that the process model is given by a linear time-varying (LTV) model of the form:

$$\boldsymbol{x}(k+1) = \boldsymbol{A}(k)\boldsymbol{x}(k) + \boldsymbol{B}(k)\boldsymbol{u}(k) \tag{1}$$

where the real plant lies within a polytope that is represented by the convex hull:

$$[\boldsymbol{A}(k)\ \boldsymbol{B}(k)] = \sum_{l=1}^{L} \beta_l [\boldsymbol{A}^{(l)}\ \boldsymbol{B}^{(l)}] \quad ; \sum_{l=1}^{L} \beta_l = 1;\ \beta_l \geq 0 \tag{2}$$

Each vertex l corresponds to a linear model obtained from linearizing a nonlinear model or identification of a linear model in the neighbourhood of a particular operating point. It is assumed that the states are fully measured. The states and the controlled and manipulated variables in model (1) can be decomposed into N subsystems as follows:

$$\begin{bmatrix} \boldsymbol{x}_{11}(k+1) \\ \vdots \\ \boldsymbol{x}_{ii}(k+1) \\ \vdots \\ \boldsymbol{x}_{NN}(k+1) \end{bmatrix} = \begin{bmatrix} \boldsymbol{A}_{11}(k) & \cdots & \cdots & \cdots & \boldsymbol{A}_{1N}(k) \\ \vdots & \ddots & & & \vdots \\ \boldsymbol{A}_{i1}(k) & \cdots & \ddots & \cdots & \boldsymbol{A}_{iN}(k) \\ \vdots & & & \ddots & \vdots \\ \boldsymbol{A}_{N1}(k) & \cdots & \cdots & \cdots & \boldsymbol{A}_{NN}(k) \end{bmatrix} \begin{bmatrix} \boldsymbol{x}_{11}(k) \\ \vdots \\ \boldsymbol{x}_{ii}(k) \\ \vdots \\ \boldsymbol{x}_{NN}(k) \end{bmatrix} + \begin{bmatrix} \boldsymbol{B}_{11}(k) & \cdots & \cdots & \cdots & \boldsymbol{B}_{1N}(k) \\ \vdots & \ddots & & & \vdots \\ \boldsymbol{B}_{i1}(k) & \cdots & \ddots & \cdots & \boldsymbol{B}_{iN}(k) \\ \vdots & & & \ddots & \vdots \\ \boldsymbol{B}_{N1}(k) & \cdots & \cdots & \cdots & \boldsymbol{B}_{NN}(k) \end{bmatrix} \begin{bmatrix} \boldsymbol{u}_1(k) \\ \vdots \\ \boldsymbol{u}_i(k) \\ \vdots \\ \boldsymbol{u}_N(k) \end{bmatrix} \tag{3}$$

where $i \in \{1,\cdots,N\}$; $\boldsymbol{x}_{ii} \in \Re^{n_{ii}}$; $\boldsymbol{u}_i \in \Re^{m_i}$. For example, in model (3) the i[th] controller for the i[th] subsystem is based on the following model:

$$\boldsymbol{x}_i(k+1) = \boldsymbol{A}_i(k)\boldsymbol{x}_i(k) + \boldsymbol{B}_i(k)\boldsymbol{u}_i(k) + \sum_{\substack{j=1 \\ j\neq i}}^{N} \boldsymbol{B}_j(k)\boldsymbol{u}_j(k) \tag{4}$$

and similar to the representation given in (2) it is assumed that for the i[th] subsystem (4):

$$[\boldsymbol{A}_i(k)\ \boldsymbol{B}_i(k)..\ \boldsymbol{B}_j(k)..] = \sum_{l=1}^{L} \beta_l [\boldsymbol{A}_i^{(l)}\ \boldsymbol{B}_i^{(l)}..\ \boldsymbol{B}_j^{(l)}..] \quad \forall j \in \{1,...,N\},\ j \neq i \tag{5}$$

where $\boldsymbol{x}_i' = \left[\boldsymbol{x}_{11}', \cdots\cdots, \boldsymbol{x}_{ii}', \cdots\cdots, \boldsymbol{x}_{NN}'\right]'$ is the vector of states of subsystem i containing states $\boldsymbol{x}_{ii}$ that can be measured locally augmented with states $\boldsymbol{x}_{jj}$ that affect subsystem i measured in the other subsystems and communicated among the subsystems. Therefore the matrix $\boldsymbol{A}_i(k)$ contains all the elements of the matrix $\boldsymbol{A}(k)$. Model (4) also includes the effect of local controller $\boldsymbol{u}_i$ and the other controllers $\boldsymbol{u}_j$ with their corresponding matrices defined as:

$$\begin{aligned} \boldsymbol{B}_i'(k) &= \left[\boldsymbol{B}_{1i}'(k), \cdots\cdots, \boldsymbol{B}_{Ni}'(k)\right]' \\ \boldsymbol{B}_j'(k) &= \left[\boldsymbol{B}_{1j}'(k), \cdots\cdots, \boldsymbol{B}_{Nj}'(k)\right]' \end{aligned} \tag{6}$$

The model in (4) is general and can be used to represent special limiting cases such as the decentralized case where all the interactions are ignored, e.g. $\boldsymbol{B}_j' = [\boldsymbol{0}]\ \forall j \in \{1,...,N\},\ j \neq i$.

2.2 Robust Performance Objective

Kothare *et al.* (1996) proposed a formulation for a centralized problem whereby an upper bound on a robust performance objective is minimized. In the current work a similar formulation is used but the minimization is simultaneously done for every subsystem i defined by (4) for which the following min-max problem is solved:

$$\begin{aligned} &\min_{\boldsymbol{u}_i(k+n|k)} \ \max_{[\boldsymbol{A}_i(k+n)\ \boldsymbol{B}_i(k+n)\ \boldsymbol{B}_j(k+n)],n\geq 0} J_i(k) \\ &s.t. \\ &\left|\boldsymbol{u}_i(k+n|k)\right| \leq \boldsymbol{u}_i^{max},\ n \geq 0 \end{aligned} \tag{7}$$

In general, the local objective $J_i(k)$ is defined as follows:

$$\begin{aligned} J_i(k) = \sum_{n=0}^{\infty} [&\boldsymbol{x}_i'(k+n|k)\mathbb{Q}_i\boldsymbol{x}_i(k+n|k) \\ &+ \boldsymbol{u}_i'(k+n|k)\boldsymbol{R}_i\boldsymbol{u}_i(k+n|k) \\ &+ \sum_{\substack{i=1 \\ i\neq j}}^{N} \boldsymbol{u}_j^{\bullet\prime}(k+n|k)\boldsymbol{R}_j\boldsymbol{u}_j^{\bullet}(k+n|k)] \end{aligned} \tag{8}$$

where $\mathbb{Q}_i > 0$, $\boldsymbol{R}_i > 0$, $\boldsymbol{R}_j > 0$. The local objective given in (8) takes into account the goals of the other controllers, third summation in the RHS, in order to achieve the global objective of the entire system. The superscript "$\bullet$" indicates that the solution was obtained in a previous iteration and remains fixed in the current iteration as will be explained later. It should be pointed out that one can easily modify the problem in (8) to solve particular objectives such as Nash equilibrium or decentralized control. Both strategies are based on minimizing strictly local objectives of the subsystems. The difference is that for Nash the interaction information is shared among the subsystems while for decentralized control the interaction information is neglected. Accordingly, for both Nash and decentralized control $\mathbb{Q}_i$ and $\boldsymbol{R}_i$ in (8) are modified to contain all zeros except for the weights corresponding to the local subsystem and the third summation in the RHS of (8) is excluded. On the other hand the interaction term in (4) is included for Nash but it is ignored for decentralized control.

Since the objective in (8) has an infinite horizon, the problem of finding infinite $\boldsymbol{u}_i$ is computationally intractable. Instead, a state-feedback law is sought for each subsystem i as follows:

$$\begin{aligned} \boldsymbol{u}_i(k+n|k) &= \boldsymbol{F}_{ii}\boldsymbol{x}_{ii}(k+n|k) + \sum_{\substack{j=1 \\ j\neq i}}^{N} \boldsymbol{F}_{ij}\boldsymbol{x}_{ij}(k+n|k) \\ &= \boldsymbol{F}_i\boldsymbol{x}_i(k+n|k) \end{aligned} \tag{9}$$

similarly,

$$\begin{aligned} \boldsymbol{u}_j^{\bullet}(k+n|k) &= \boldsymbol{F}_{jj}^{\bullet}\boldsymbol{x}_{jj}(k+n|k) + \sum_{\substack{i=1 \\ i\neq j}}^{N} \boldsymbol{F}_{ji}^{\bullet}\boldsymbol{x}_{ji}(k+n|k) \\ &= \boldsymbol{F}_j^{\bullet}\boldsymbol{x}_i(k+n|k) \end{aligned} \tag{10}$$

Using these state-feedback laws in (4) leads to the following closed loop model:

$$\boldsymbol{x}_i(k+1)=\left(\tilde{\boldsymbol{A}}_i(k)+\boldsymbol{B}_i(k)\boldsymbol{F}_i\boldsymbol{u}_i(k)\right)\boldsymbol{x}_i(k) \tag{11}$$

where $\tilde{A}_i(k)=A_i(k)+\sum_{\substack{j=1\\ j\neq i}}^{N}B_j(k)F_j^{\bullet}$

It is assumed that there exists a quadratic function $V_i(k)=\boldsymbol{x}_i'(k)\boldsymbol{P}_i\boldsymbol{x}_i(k)$, $\boldsymbol{P}_i>0$, so that, for any plant in (6), this function satisfies the following stability constraint:

$$\begin{aligned}V_i(k+n+1|k)-V_i(k+n|k)\le -[&\boldsymbol{x}_i'(k+n|k)\mathbb{Q}_i\boldsymbol{x}_i(k+n|k)\\&+\boldsymbol{u}_i'(k+n|k)\boldsymbol{R}_i\boldsymbol{u}_i(k+n|k)\\&+\sum_{\substack{i=1\\ i\neq j}}^{N}\boldsymbol{u}_j^{\bullet\prime}(k+n|k)\boldsymbol{R}_j\boldsymbol{u}_j^{\bullet}(k+n|k)]\\ n\ge 0\end{aligned} \tag{12}$$

Using (11), the robust stability constraint in (12) becomes:

$$\begin{aligned}V_i(k+n+1|k)-V_i(k+n|k)\le -[&\boldsymbol{x}_i'(k+n|k)\tilde{\mathbb{Q}}_i\boldsymbol{x}_i(k+n|k)\\&+\boldsymbol{u}_i'(k+n|k)\boldsymbol{R}_i\boldsymbol{u}_i(k+n|k)]\end{aligned}$$

where $\tilde{\mathbb{Q}}_i=\mathbb{Q}_i+\sum_{\substack{i=1\\ i\neq j}}^{N}\boldsymbol{F}_j^{\bullet\prime}(k+n|k)\boldsymbol{R}_j\boldsymbol{F}_j^{\bullet}(k+n|k)$ (13)

which, for all $n\ge 0$, turns out to be:

$$\begin{aligned}\left[\tilde{\boldsymbol{A}}_i(k+n)+\boldsymbol{B}_i(k+n)\boldsymbol{F}_i\right]'\boldsymbol{P}_i\left[\tilde{\boldsymbol{A}}_i(k+n)+\boldsymbol{B}_i(k+n)\boldsymbol{F}_i\right]&\\ -\boldsymbol{P}_i+\boldsymbol{F}_i'\boldsymbol{R}_i\boldsymbol{F}_i+\tilde{\mathbb{Q}}_i\le 0&\end{aligned} \tag{14}$$

By defining an upper bound, i.e.

$$J_i(k)\le \boldsymbol{x}_i'(k)\boldsymbol{P}_i\boldsymbol{x}_i(k)=V_i(k)\le\gamma_i \tag{15}$$

and substituting the parameterization $\boldsymbol{F}_i=\boldsymbol{Y}_i'\boldsymbol{Q}_i^{-1}$, $\boldsymbol{Q}_i=\gamma_i\boldsymbol{P}_i^{-1}$, followed by performing Schur complements (Boyd et al. 1994) on (14) and (15) it can be easily shown that the minimization of $J_i(k)$ is equivalent to the minimization of its upper bound γ_i as in the following linear minimization problem with LMI constraints (Kothare *et al.* 1996):

$$\begin{aligned}&\min_{\gamma_i,\boldsymbol{Q}_i,\boldsymbol{Y}_i}\ \gamma_i\\ &s.t.\\ &\begin{bmatrix}1 & \boldsymbol{x}_i'(k)\\ \boldsymbol{x}_i(k) & \boldsymbol{Q}_i\end{bmatrix}\ge 0\\ &\begin{bmatrix}\boldsymbol{Q}_i & \boldsymbol{Q}_i\tilde{\boldsymbol{A}}_i^{\prime(l)}+\boldsymbol{Y}_i'\boldsymbol{B}_i^{\prime(l)} & \boldsymbol{Q}_i\tilde{\mathbb{Q}}_i^{1/2} & \boldsymbol{Y}_i'\boldsymbol{R}_i^{1/2}\\ * & \boldsymbol{Q}_i & \boldsymbol{0} & \boldsymbol{0}\\ * & * & \gamma_i\boldsymbol{I} & \boldsymbol{0}\\ * & * & * & \gamma_i\boldsymbol{I}\end{bmatrix}\ge 0\\ &\qquad\forall l\in\{1,\dots,L\}\\ &\begin{bmatrix}\left(\boldsymbol{u}_i^{max}\right)^2\boldsymbol{I} & \boldsymbol{Y}_i\\ \boldsymbol{Y}_i' & \boldsymbol{Q}_i\end{bmatrix}\ge 0\end{aligned} \tag{16}$$

The key difference between the centralized control algorithm proposed by Kothare *et al.* (1996) and the distributed strategy proposed in this work is that every controller in the set $i\in\{1,\cdots,N\}$ solves a local problem as in (16) and then the solutions are exchanged in an iterative scheme that is further explained in the next subsection. It should be remembered that one of the key reasons to use distributed MPC strategies is to address real time computation issues when dealing with large-scale processes (Li *et al.* 2005). Although the proposed iterative scheme tends to increase the computational time, the problem defined in (16) is numerically advantageous as compared to solving the same problem for the whole system (centralized control). The reason is that the state feedback controller for each subsystem i is obviously of smaller dimensions than a state feedback controller of the centralized MPC strategy. For instance, $\boldsymbol{Y}_i$ for subsystem i is of dimension $(n\times m_i)$ instead of $(n\times m)$ for centralized system where m is the total number of manipulated variables of the entire process.

2.3 Robust DMPC Algorithm

This section presents the main result of the paper where an on-line algorithm for RDMPC is proposed. It is assumed that there is an ideal communication network available so that the controllers can exchange their information with no delays. The goal of performing communication and exchanging solutions among controllers is to achieve the optimal solution of the entire system in an iterative fashion. The algorithm proceeds according to the Jacobi iteration method used for the solution of systems of algebraic equations. The procedure is summarized in *Algorithm 1* below.

Algorithm 1 (RDMPC)

Step0 (initialization)*: at control interval k=0 set* $\boldsymbol{F}_i=0$.

Step1 (updating) *at control interval (k) all the controllers exchange their local states measurements and initial estimates* $\boldsymbol{F}_i$*'s via communication, set iteration t = 0 and* $\boldsymbol{F}_i=\boldsymbol{F}_i^{(0)}$.

Step2 (iterations)

while $t\le t_{max}$

Solve all N LMI problems (16) in parallel to obtain the minimizers $\boldsymbol{Y}_i^{(t+1)},\boldsymbol{Q}_i^{(t+1)}$ *to estimate the feedback solutions* $\boldsymbol{F}_i^{(t+1)}=\boldsymbol{Y}_i'^{(t+1)}\boldsymbol{Q}_i^{-1(t+1)}$. *If problem is infeasible set* $\boldsymbol{F}_i^{(t)}=\boldsymbol{F}_i^{(t-1)}$. *Check the convergence for a specified error tolerance* ε_i *for all the controllers*

if $\left\|\boldsymbol{F}_i^{(t+1)}-\boldsymbol{F}_i^{(t)}\right\|\le\varepsilon_i\ \ \forall i\in\{1,\dots,N\}$

break

end if

Exchange the solutions ($\boldsymbol{F}_i$*'s) and set t = t + 1*

end while

Step3 (implementation) *apply the control actions* $\boldsymbol{u}_i = \boldsymbol{F}_i\boldsymbol{x}_i$ *to the corresponding subsystems, increase the control interval* $k = k + 1$*, return to step1 and repeat the procedure.*

Algorithm1 is implemented in MATLAB® and problem (16) is solved via MATLAB® LMI solver. Convergence of the iterations in Step 2 and stability properties are discussed in the following subsection.

2.4 Convergence and Robust Stability Analysis of RDMPC Algorithm

Regarding convergence, it can be shown that at each time interval, each one of the *N* convex problems defined in *Algorithm1* will converge to the same solution which is the solution of the centralized problem, i.e. $\gamma_1 = \gamma_2 = \cdots = \gamma_i = \cdots = \gamma_N = \gamma$ where γ is the performance upper bound of centralized MPC. For brevity, a two subsystem situation, i.e. *N*=2 is considered without loss of generality. It is also assumed that the solutions are feasible.

Define:

$$\text{for subsystem 1} \quad \gamma_1^{(t)} = \min_{\boldsymbol{F}_1^{(t)}} \gamma_1(\boldsymbol{F}_1^{(t)}, \boldsymbol{F}_2^{(t-1)})$$

$$\text{for subsystem 2} \quad \gamma_2^{(t)} = \min_{\boldsymbol{F}_2^{(t)}} \gamma_2(\boldsymbol{F}_1^{(t-1)}, \boldsymbol{F}_2^{(t)})$$

$$\text{Then,} \quad \gamma_1^{(t)} \le \gamma_2^{(t-1)} \tag{a}$$

and the reason being that both sides of this inequality are using the same value of $\boldsymbol{F}_2 = \boldsymbol{F}_2^{(t-1)}$ but the LHS minimizes γ with respect to $\boldsymbol{F}_1$ whereas the RHS of the inequality uses a not necessarily optimal value of $\boldsymbol{F}_1 = \boldsymbol{F}_1^{(t-1)}$. Following the same argument:

$$\gamma_2^{(t)} \le \gamma_1^{(t-1)} \tag{b}$$

Thus, the γ_i's decrease until (a) or (b) become equalities. Since the minimizations are convex and lead to global optimal solutions, this occurs only when $\boldsymbol{F}_1^{(t)} = \boldsymbol{F}_1^{(t-1)}$ and $\boldsymbol{F}_2^{(t)} = \boldsymbol{F}_2^{(t-1)}$ and consequently $\gamma_{sub1}^{(t)} = \gamma_{sub2}^{(t)} = \gamma$, i.e. the minimization with respect to both $\boldsymbol{F}_1$ and $\boldsymbol{F}_2$ give the same solution which must be, following convexity of problem (16), equal to the global optimum of the centralized control problem that has an identical formulation to (16). The robust stability of *Algorithm1* follows from the fact that for each subsystem, a robust stability related constraint is enforced by one of the linear matrix inequalities in problem (16). Thus each one of the *N* controllers satisfies robust stability. Although theoretical convergence of the Jacobi iteration can be proven, it was found that numerical noise exists due to inaccuracies of the LMI solvers in obtaining the solution of problem (16). Consequently, to speed up convergence in the presence of this numerical noise when *Algorithm1* is implemented, the *successive Relaxation* (SR) method is employed (Hageman and Young 1981). The SR method is applied to the solution obtained from (16) for each subsystem to estimate a weighted average between the current and previous iterate solutions. The method is given by the following recurrence formula:

$$\boldsymbol{F}_i^{(t+1)} = \alpha\bar{\boldsymbol{F}}_i^{(t+1)} + (1-\alpha)\boldsymbol{F}_i^{(t)} \tag{17}$$

where α is a parameter to be specified by the user in order to accelerate convergence. $\bar{\boldsymbol{F}}_i^{(t+1)}$ denotes the solution obtained at the current iteration from (16) whereas $\boldsymbol{F}_i^{(t+1)}$ is the estimate to be used in the next iteration. Typically, α can be chosen from values between 0 and 2 and when it is set to 1 the normal iterative scheme is retrieved. Since there is no systematic way to select a value for α in advance, simulations with different values of α have to be performed as shown in the first example.

3. Case Studies

3.1 Example 1

A distillation column control problem studied by Venkat (2006) is considered with the difference that uncertainties in the steady-state gains of the model are added to illustrate the robustness of the proposed algorithm. Accordingly, the real model lies within a polytope defined within the two vertices:

$$G_1 = \begin{bmatrix} \dfrac{32.63}{(99.6s+1)(0.35s+1)} & \dfrac{-33.89}{(98.02s+1)(0.42s+1)} \\ \dfrac{34.84}{(110.5s+1)(0.03s+1)} & \dfrac{-18.85}{(75.43s+1)(0.3s+1)} \end{bmatrix}$$

$$G_2 = \begin{bmatrix} \dfrac{326.3}{(99.6s+1)(0.35s+1)} & \dfrac{-338.9}{(98.02s+1)(0.42s+1)} \\ \dfrac{348.4}{(110.5s+1)(0.03s+1)} & \dfrac{-188.5}{(75.43s+1)(0.3s+1)} \end{bmatrix} \tag{21}$$

A state-space model, not shown for brevity, is obtained from a canonical realization of equation (21). To demonstrate the effectiveness of the proposed method the bad pairings, according to the *Relative Gain Array* RGA, are selected, i.e. the RGA element λ_{11} is -1.0874 and accordingly the "bad" pairings are u_1-y_1 (*subsystem1*) and u_2-y_2 (*subsystem2*). The physical constraints on manipulated variables are given by:

$$|u_1(k+n)| \le 1.5; \ |u_2(k+n)| \le 2; \ n \ge 0 \tag{22}$$

For the purpose of comparison between different cases, a cost function is defined as follows:

$$J_{cost} = (1/2Ns)\sum_{j=0}^{Ns}\sum_{i=1}^{N}\left(\boldsymbol{x}_i'(j)\mathbb{Q}_i\boldsymbol{x}_i(j) + \boldsymbol{u}_i'(j)\boldsymbol{R}_i\boldsymbol{u}_i(j)\right) \tag{23}$$

where *Ns* is the simulation time. The following parameters are used for the two controllers: $\mathbb{Q}_{y1} = \mathbb{Q}_{y2} = 50$ so that $\mathbb{Q}_i = \boldsymbol{C}_i'\mathbb{Q}_{yi}\boldsymbol{C}_i + 10^{-6}\boldsymbol{I}$ where $\boldsymbol{C}_i$ is the measurement matrix such that $\boldsymbol{y}_i = \boldsymbol{C}_i\boldsymbol{x}_i$; $\boldsymbol{R}_1 = \boldsymbol{R}_2 = 1$; $\alpha = 0.95$. The value of α is selected, as mentioned above, based on simulations by trial and error to speed convergence of the Jacobi iteration. The number of iterations that was required to satisfy the convergence criteria of *Algorithm1* for different values of α

is given in Table1. α=*0.95* resulted in the fastest convergence.

Table 1. Effect of α on convergence with $\varepsilon_1=\varepsilon_2=10^{-3}$

α	# iterations
1.05	55
1.00	38
0.95	28
0.90	32
0.8	38

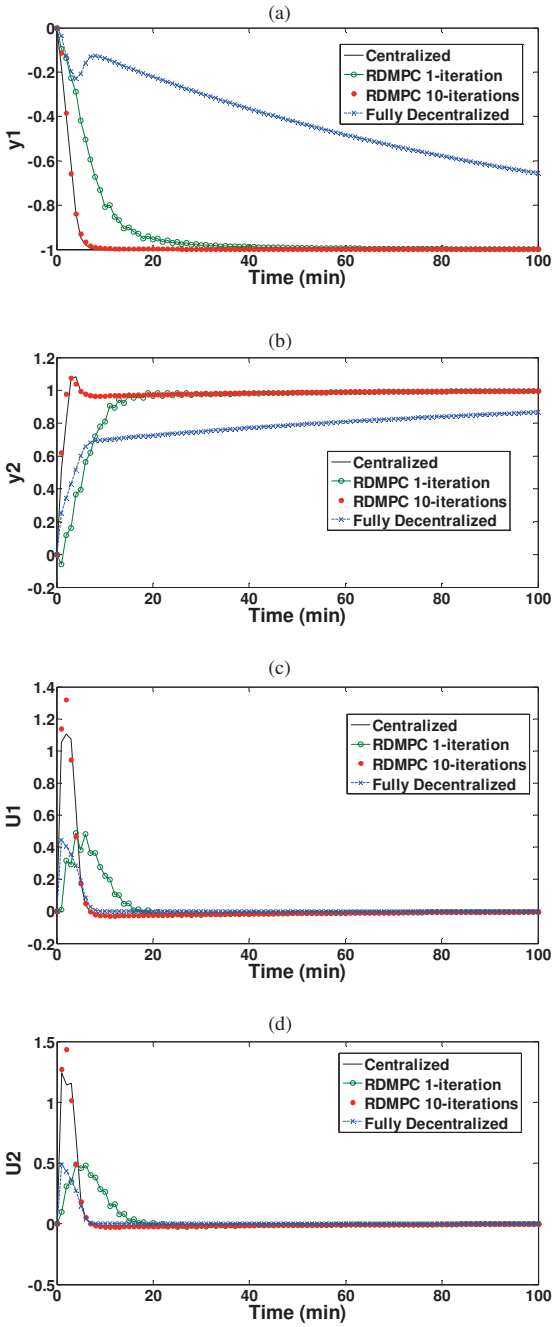


Fig. 1. Dynamic response in controlled and manipulated variables for set-point changes in y1 and y2.

Three cases are considered for the application of *Algorithm1*; fully decentralized, RDMPC with one iteration, and RDMPC with 10 iterations. It should be remembered that as indicated in section 2, the cost γ decreases monotonically with the number of iterations. Thus, even after one iteration, a performance improvement is expected. The motivation for using a small number of iterations, as mentioned earlier, is to use distributed MPC strategies to address real time computation issues when dealing with large scale processes. The decentralized strategy used in this study is obtained, as explained in Section 2.2, with *Algorithm1* by ignoring interactions in equation (4). Then, the performance of *Algorithm1* with these 3 different schemes was compared to the centralized strategy in Figure1. The simulations correspond to simultaneous changes in set-points of both controlled variables y1 and y2 by -1 and 1; respectively.

In comparison with the centralized scheme, the performance of RDMPC approaches that of the centralized scheme as the number of iterations is increased. The fully decentralized case resulted as expected in the worst performance. A comparison of the cost in (23) for different schemes is given in Table2. This table illustrates that *Algorithm1* can be used, depending on the chosen number of iterations, to obtain a performance that varies between two extremes corresponding to the fully decentralized and the centralized strategies; respectively. It is also clear, from figures 1(c) and 1(d), that the constraints given in (22) are satisfied.

Table 2. Cost for different strategies (example1)

Strategy	Cost (23)
Centralized	0.92
RDMPC (10 iteration)	0.93
RDMPC (1 iteration)	2.43
Fully decentralized	35.9

3.2 Example 2

This example considers the high-purity column originally studied by Skogestad and Morari (1988). The nominal transfer function of this system is given by:

$$\begin{bmatrix} y_1 \\ y_2 \end{bmatrix} = \frac{1}{75s+1}\begin{bmatrix} 0.878 & 0.864 \\ 1.082 & 1.096 \end{bmatrix}\begin{bmatrix} u_1 \\ u_2 \end{bmatrix} \quad (24)$$

A state-space model is obtained based on a canonical realization of equation (24) and not shown for brevity. Due to the high condition number, this process has been used in the past to illustrate closed-loop sensitivity to model errors. The model uncertainty is given by errors in steady-state gains. The gains in the first column of the transfer matrix in (24) are expected to change by up to +80% whereas the gains in the second column are expected to change by up to -80%. The constraints on manipulated variables are represented by $|\boldsymbol{u}(k+n)| \le 1$, $n \ge 0$. The system given above was decomposed into two subsystems; *viz.*, y_1-u_1 (*subsystem1*) and y_2-u_2 (*subsystem2*). The controllers parameters used in simulation are; $\mathbb{Q}_1 = \mathbb{Q}_2 = 1$, $R_1 = R_2 = 1$, $\alpha = 1$, $\varepsilon_1 = \varepsilon_2 = 10^{-2}$.

Figure 2 depicts the performance of *Algorithm1* compared with centralized MPC for a unit set-point change in y_1 and it illustrates that RDMPC algorithm results in an identical response as the centralized MPC. For this example, the RDMPC algorithm converges very quickly in about three iterations after which the error tolerances specified above ($\varepsilon_1=\varepsilon_2=10^{-2}$) are met. Figure 3 shows the convergent behaviour of the RDMPC algorithm obtained in the first sampling interval. The upper bounds γ_1 and γ_2 for subsystems 1 and 2 respectively, obtained by solving (16) in parallel and by applying *Algorithm1*, converge to the same value after about 3 iterations and this value is identical to that obtained for centralized MPC. The cost, defined by equation (23), for both strategies, is equal to *7.48*. To show the ability of the method to deal with different objective functions an RDMPC with a Nash equilibrium objective and a robust decentralized MPC were designed by proper choice of the weights $\mathbb{Q}_i$ and $\boldsymbol{R}_i$ as explained in section 2.2. The results with the Nash-equilibrium based controller, shown also in Figure 2, are similar to the centralized case and the cost was *7.55*, slightly larger than the centralized MPC cost. The decentralized MPC, not shown in the Figure, resulted as expected in a slightly higher cost than Nash of *7.73*.

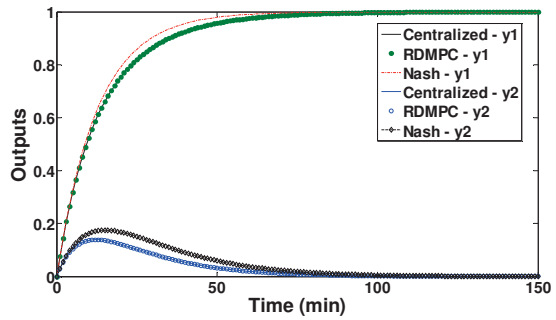


Fig. 2. Dynamic response to unit set-point change in y1.

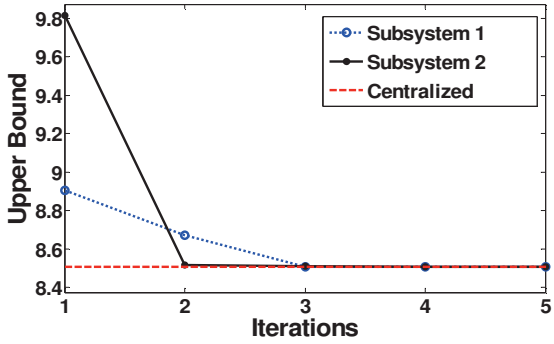


Fig. 3. Convergence characteristics of *Algorithm1* at the first sampling time.

4. CONCLUSIONS

The main goal of this work was to propose an on-line algorithm for DMPC strategy that explicitly considers model errors. The main idea of the proposed method is to decompose the model of the whole system into *N* subsystems and then obtain a local state feedback controller by minimizing an upper bound on a robust performance objective for each subsystem. The subsystem performance takes into account the objectives of the other subsystems in order to achieve the goal of the entire system. The method was also suitable for pursuing other objectives such as Nash equilibrium or decentralized control in the presence of model errors. The problem was converted into *N* convex problems with linear matrix inequalities and solved iteratively by using the Jacobi iteration method with successive relaxation (SR). Although convergence of the iterative solution was proven, the SR feature was helpful for filtering numerical noise in the LMI solutions resulting in faster convergence. When convergence was reached, the algorithm led to the same solution of the centralized MPC problem. The examples showed that RDMPC can achieve, after a sufficient number of iterations, equivalent performance to centralized control. Moreover, the examples illustrated that improvements in RDMPC performance as compared to decentralized control can be achieved with a relatively small number of iterations.

ACKNOWLEDGMENT

The first author would like to thank the General Secretariat of Higher Education of Libya for financial support. Also, the authors acknowledge additional support of Natural Sciences and Engineering Research Council of Canada (NSERC).

REFERENCES

Boyd, S., El.Ghaoui, L., Feron, E., and Balakrishnan, V. (1994). *Linear matrix inequalities in system and control theory*, SIAM, USA.

Hageman, L. and Young, D. (1981) *Applied iterative methods.* Academic Press, NY.

Kothare, M. V., Balakrishnan, V., and Morari, M. (1996) Robust constrained model predictive control using linear matrix inequalities. *Automatica*, **32** (10), 1361-1379.

Li, S., Zhang, Y., and Zhu, Q. (2005). Nash-optimization enhanced distributed model predictive control applied to the shell benchmark problem. *Information science,* **170,** 329-349.

Qin, S. J. and Badgwell, T. A. (2003). A survey of industrial model predictive control technology. *Control engineering practice,* **11,** 733-764.

Rawlings, James B. and Stewart, Brett T. (2008). Coordinating multiple optimization-based controllers: new opportunities and challenges. *Journal of Process Control,* **18,** 839-845.

Skogestad, S. and Morari, M. (1988). Robust control of ill-conditioned plants: high-purity distillation. *IEEE transactions on automatic control,* **12,** 1092-1104.

Venkat, A. N. (2006). *Distributed model predictive control: theory and applications*, PhD thesis, USA.

Zhang, Y., and Li, S. (2007). Networked model predictive control based on neighbourhood optimization for serially connected large-scale processes. *Journal of Process Control,* **17,** 37-50.

.

Advances in Modeling, Estimation, and Identification

Poster Session

Multirefinery and Petrochemical Networks Design and Integration

K. Al-Qahtani*, A. Elkamel**
E. Alper***

** Saudi Aramco, Dhahran, Saudi Arabia 31311*
(Tel.: (966)-875-4828. e-mail: Khalid.qahtani.7@aramco.com)
*** Department of Chemical Engineering, University of Waterloo, Ontario, Canada N2L 3G1*
(Tel.: (519)-888-4567 ext. 37157. e-mail: aelkamel@uwaterloo.ca)
**** Department of Chemical Engineering, Hacettepe University, Beytepe, Turkey 06800*
((Tel.: (312)-297-7400. e-mail: ealper@hacettepe.edu.tr)

Abstract: In this paper we propose a model for the design of an optimal network integration of multisite refinery and petrochemical systems under uncertainty. The proposed model was formulated as a two-stage stochastic mixed-integer problem with the objective of minimizing the refining cost over a given time horizon and maximizing the added value by the petrochemical network. Uncertainties considered in this study were in terms of imported crude oil price, refinery product price, petrochemical product price, refinery market demand, and petrochemical lower level product demand. The proposed method adopts the sample average approximation (SAA) method for scenario generation and optimal gap statistical bounding. The model performance was tested on an industrial case study of multiple refineries and a polyvinyl chloride (PVC) complex.

Keywords: Integration, petrochemical planning, multirefinery optimization, planning under uncertainty

1. INTRODUCTION

Process integration in the refining and petrochemical industry includes many intuitively recognized benefits of processing higher quality feedstocks, improving value of byproducts, and achieving better efficiencies through sharing of resources. This is evidently seen from the current projects around the world for building integrated refineries and the development of complex petrochemical industries that are aligned through advanced integration platforms.

Despite the fact that petroleum refining and petrochemical companies have recently engaged in more integration projects, relatively little research in the open literature have been reported mostly due to confidentiality reasons. Such concerns render the development of a systematic framework of network integration and coordination difficult. Pervious research in the field assumed either no limitations on refinery feedstock availability for the petrochemical planning problem or fixed the refinery production levels assuming an optimal operation. In this paper, we present a mathematical model for the determination of the optimal integration and coordination strategy for a refinery network and synthesize the optimal petrochemical network required to satisfy a given demand from any set of available technologies. Therefore, achieving a global optimal production strategy by allowing appropriate trade offs between the refinery and the downstream petrochemical markets. The refinery and petrochemical systems were modeled as MILP problems that will also lead to an overall refinery and petrochemical process production levels and detailed blending levels at each refinery site. Furthermore, we apply the sample average approximation (SAA) method within an iterative scheme to generate the required scenarios. The solution quality is then statistically evaluated by measuring the optimality gap of the final solution.

2. MODEL FORMULATION

The proposed formulation addresses the problem of the simultaneous design of an integrated network of refineries and petrochemical processes. The proposed model is based on the formulations proposed by Al-Qahtani and Elkamel (2008) and Al-Qahtani et al. (2008). All material balances are carried out on a mass basis with the exception of refinery quality constraints of properties that only blend by volume where volumetric flowrates are used instead. Uncertainty was accounted for using two-stage stochastic programming with recourse approach. Parameters uncertainties considered in this study included uncertainties in the imported crude oil price $CrCost_{cr}$, refinery product price Pr_{cfr}^{Ref}, petrochemical product price Pr_{cp}^{Pet}, refinery market demand $D_{Ref_{cfr}}$, and petrochemical lower level product demand $D_{Pet_{cp}}^{L}$. Uncertainty is modeled through the use of mutually exclusive scenarios of the model parameters with a finite number N of outcomes. For each $\xi_k = (CrCost_{cr,k}, Pr_{cfr,k}^{Ref}, Pr_{cp,k}^{Pet}, D_{Ref_{cfr,k}}, D_{Pet_{cp,k}}^{L})$ where $k = 1,2,...,N$, there corresponds a probability p_k. The generation of the scenarios will be briefly explained in a later section. The proposed stochastic model is as follows:

$$Min \sum_{cre\in CR}\sum_{i\in I}\sum_{k\in N} p_k\ CrCost_{cr,k}\ S^{Ref}_{cr,i} + \sum_{p\in P}\sum_{cre\in CR}\sum_{i\in I} z_{cr,p,i}\ OpCost_p$$
$$+ \sum_{cir\in CIR}\sum_{i\in I}\sum_{i'\in I} InCost_{i,i'}\ ypipe^{Ref}_{cir,i,i'} + \sum_{i\in I}\sum_{m\in M_{Ref}}\sum_{s\in S} InCost_{m,s}\ yexp^{Ref}_{m,i,s}$$
$$- \sum_{cfr\in PEX}\sum_{i\in I}\sum_{k\in N} p_k\ Pr^{Ref}_{cfr,k}\ e^{Ref}_{cfr,i} - \sum_{cp\in CP}\sum_{m\in M_{Pet}}\sum_{k\in N} p_k\ Pr^{Pet}_{cp,k}\ \delta_{cp,m}\ x^{Pet}_m \quad (1)$$
$$+ \sum_{cfr\in CFR}\sum_{k\in N} p_k\ C^{Ref+}_{cfr}\ V^{Ref+}_{cfr,k} + \sum_{cfr\in CFR}\sum_{k\in N} p_k\ C^{Ref-}_{cfr}\ V^{Ref-}_{cfr,k}$$
$$+ \sum_{cp\in CFP}\sum_{k\in N} p_k\ C^{Pet+}_{cp}\ V^{Pet+}_{cp,k} + \sum_{cp\in CFP}\sum_{k\in N} p_k\ C^{Pet-}_{cp}\ V^{Pet-}_{cp,k}$$

Subject to

$$z_{cr,p,i} = S^{Ref}_{cr,i} \quad \forall\ cr\in CR,\ i\in I \text{ and } p\in P' = \{\text{Set of CDU processes} \forall \text{ plant i}\} \quad (2)$$

$$\sum_{p\in P}\alpha_{cr,cir,i,p}\ z_{cr,p,i} + \sum_{i'\in I}\sum_{p\in P}\xi_{cr,cir,i',p,i}\ xi^{Ref}_{cr,cir,i',p,i} - Fi^{Pet}_{cr,cir\in RPI,i} - \sum_{i'\in I}\sum_{p\in P}\xi_{cr,cir,i,p,i'}\ xi^{Ref}_{cr,cir,i,p,i'} - \sum_{cfr\in CFR} w_{cr,cir,cfr,i} - \sum_{rf\in FUEL} w_{cr,cir,rf,i} = 0 \quad \forall\ cr\in CR,\ cir\in CIR,\ i'\&i\in I \text{ where } i\neq i' \quad (3)$$

$$\sum_{cre\in CR}\sum_{cir\in CB} w_{cr,cir,cfr,i} - \sum_{cre\in CR}\sum_{rf\in FUEL} w_{cr,cfr,rf,i} - \sum_{cre\in CR} Ff^{Pet}_{cr,cfr\in RPF,i} = x^{Ref}_{cfr,i} \quad \forall\ cfr\in CFR,\ i\in I \quad (4)$$

$$\sum_{cre\in CR}\sum_{cir\in CB}\frac{w_{cr,cir,cfr,i}}{sg_{cr,cir}} = xv^{Ref}_{cfr,i} \quad \forall\ cfr\in CFR,\ i\in I \quad (5)$$

$$\sum_{cir\in FUEL} cv_{rf,cir,i}\ w_{cr,cir,rf,i} + \sum_{cfr\in FUEL} w_{cr,cfr,rf,i} - \sum_{p\in P}\beta_{cr,rf,i,p}\ z_{cr,p,i} = 0 \quad \forall\ cr\in CR,\ rf\in FUEL,\ i\in I \quad (6)$$

$$\sum_{cre\in CR}\sum_{cir\in CB}\left(\begin{array}{l} att_{cr,cir,q\in Qv}\dfrac{w_{cr,cir,cfr,i}}{sg_{cr,cir}} + att_{cr,cir,q\in Qw} \\ \left[w_{cr,cir,cfr,i} - \sum_{rf\in FUEL} w_{cr,cfr,rf,i} \right] \\ - \sum_{cre\in CR} Ff^{Pet}_{cr,cfr\in RPF,i} \end{array}\right) \geq q^{L}_{cfr,q\in Qv}\ xv^{Ref}_{cfr,i} + q^{L}_{cfr,q\in Qw}\ x^{Ref}_{cfr,i} \quad \forall\ cfr\in CFR,\ q=\{Qw,Qv\},\ i\in I \quad (7)$$

$$\sum_{cre\in CR}\sum_{cir\in CB}\left(\begin{array}{l} att_{cr,cir,q\in Qv}\dfrac{w_{cr,cir,cfr,i}}{sg_{cr,cir}} + att_{cr,cir,q\in Qw} \\ \left[w_{cr,cir,cfr,i} - \sum_{rf\in FUEL} w_{cr,cfr,rf,i} \right] \\ - \sum_{cre\in CR} Ff^{Pet}_{cr,cfr\in RPF,i} \end{array}\right) \leq q^{U}_{cfr,q\in Qv}\ xv^{Ref}_{cfr,i} + q^{U}_{cfr,q\in Qw}\ x^{Ref}_{cfr,i} \quad \forall\ cfr\in CFR,\ q=\{Qw,Qv\},\ i\in I \quad (8)$$

$$MinC_{m,i} \leq \sum_{p\in P}\gamma_{m,p}\sum_{cre\in CR} z_{cr,p,i} \leq MaxC_{m,i} + \sum_{s\in S} AddC_{m,i,s}\ yexp^{Ref}_{m,i,s} \quad \forall\ m\in M_{Ref},\ i\in I \quad (9)$$

$$\sum_{cre\in CR}\sum_{p\in P}\xi_{cr,cir,i,p,i'}\ xi^{Ref}_{cr,cir,i,p,i'} \leq F^{U}_{cir,i,i'}\ ypipe^{Ref}_{cir,i,i'} \quad \forall\ cir\in CIR,\ i'\&i\in I \text{ where } i\neq i' \quad (10)$$

$$\sum_{i\in I}\left(x^{Ref}_{cfr,i} - e^{Ref}_{cfr',i}\right) + V^{Ref+}_{cfr,k} - V^{Ref-}_{cfr,k} = D_{Ref_{cfr,k}} \quad \forall\ cfr\in CFR,\ cfr'\in PEX,\ k\in N \quad (11)$$

$$IM^{L}_{cr} \leq \sum_{i\in I} S^{Ref}_{cr,i} \leq IM^{U}_{cr} \quad \forall\ cr\in CR \quad (12)$$

$$Fn^{Pet}_{cp\in NRF} + \sum_{i\in I}\sum_{cre\in CR} Fi^{Pet}_{cr,cp\in RPI,i} + \sum_{i\in I}\sum_{cre\in CR} Ff^{Pet}_{cr,cp\in RPF,i} + \sum_{m\in M_{Pet}}\delta_{cp,m,k}\ x^{Pet}_m + V^{Pet+}_{cp\in CFP,k} - V^{Pet-}_{cp\in CFP,k} = D_{Pet^{L}_{cp\in CFP,k}} + xi^{Pet}_{cp\in CIP} \quad \forall\ cp\in CP,\ k\in N \quad (13)$$

$$Fn^{Pet}_{cp\in NRF} + \sum_{i\in I}\sum_{cre\in CR} Fi^{Pet}_{cr,cp\in RPI,i} + \sum_{i\in I}\sum_{cre\in CR} Ff^{Pet}_{cr,cp\in RPF,i} + \sum_{m\in M_{Pet}}\delta_{cp,m}\ x^{Pet}_m \leq D_{Pet^{U}_{cp\in CFP}} \quad \forall\ cp\in CP \quad (14)$$

$$B^{L}_m\ yproc^{Pet}_m \leq x^{Pet}_m \leq K^{U}\ yproc^{Pet}_m \quad \forall\ m\in M_{Pet} \quad (15)$$

$$\sum_{cp\in CIP} yproc^{Pet}_m \leq 1 \quad \forall\ m\in M_{Pet} \text{ that produces } cp\in CIP \quad (16)$$

$$\sum_{cp\in CFP} yproc^{Pet}_m \leq 1 \quad \forall\ m\in M_{Pet} \text{ that produces } cp\in CFP \quad (17)$$

$$Fn^{Pet}_{cp} \leq S^{Pet}_{cp} \quad \forall\ cp\in NRF \quad (18)$$

The above formulation is a two-stage stochastic mixed-integer linear programming (MILP) model. Objective function (1) represents a minimization of the annualized cost which consists of crude oil cost, refineries operating cost, refineries intermediate exchange piping cost, refinery production system expansion cost, less the refinery export revenue, added value by the petrochemical processes, plus the recourse variables of refinery and petrochemical networks; respectively. Inequality (2) corresponds to each refinery raw materials balance where throughput to each distillation unit $p\in P'$ at plant $i\in I$ from each crude type $cr\in CR$ is equal to the available supply $S_{cr,i}$. Constraint (3) represents the intermediate material balances within and

across the refineries where the coefficient $\alpha_{cr,cir,i,p}$ can assume either a positive sign if it is an input to a unit or a negative sign if it is an output from a unit. The multirefinery integration matrix $\xi_{cr,cir,i,p,i'}$ accounts for all possible alternatives of connecting intermediate streams $cir \in CIR$ of crude $cr \in CR$ from refinery $i \in I$ to process $p \in P$ in plant $i' \in I'$. The variable $xi^{Ref}_{cr,cir,i,p,i'}$ represents the transshipment flowrate of crude $cr \in CR$, of intermediate $cir \in CIR$ from plant $i \in I$ to process $p \in P$ at plant $i' \in I$. Constraint (3) also considers the petrochemical network feedstock from the refinery intermediate streams $Fi^{Pet}_{cr,cir,i}$ of each intermediate product $cir \in RPI$. The material balance of final products in each refinery is expressed as the difference between flowrates from intermediate steams $w_{cr,cir,cfr,i}$ for each $cir \in CIR$ that contribute to the final product pool and intermediate streams that contribute to the fuel system $w_{cr,cfr,rf,i}$ for each $rf \in FUEL$ less the refinery final products $Ff^{Pet}_{cr,cfr,i}$ for each $cfr \in RPF$ that are fed to the petrochemical network as shown in constraint (4). In constraint (5) we convert the mass flowrate to volumetric flowrate by dividing it by the specific gravity $sg_{cr,cir}$ of each crude type $cr \in CR$ and intermediate stream $cir \in CB$. This is needed in order to express the quality attributes that blend by volume in blending pools. Constraint (6) is the fuel system material balance where the term $cv_{rf,cir,i}$ represents the caloric value equivalent for each intermediate $cir \in CB$ used in the fuel system at plant $i \in I$. The fuel production system can either consist of a single or combination of intermediates $w_{cr,cir,rf,i}$ and products $w_{cr,cfr,rf,i}$. The matrix $\beta_{cr,rf,i,p}$ corresponds to the consumption of each processing unit $p \in P$ at plant $i \in I$ as a percentage of unit throughput. Constraints (7) and (8), respectively, represent a lower and an upper bounds on refinery quality constraints for all refinery products that either blend by mass $q \in Q_w$ or by volume $q \in Q_v$. Constraint (9) represents the maximum and minimum allowable flowrate to each processing unit. The coefficient $\gamma_{m,p}$ is a zero-one matrix for the assignment of production unit $m \in M_{Ref}$ to process operating mode $p \in P$. The term $AddC_{m,i,s}$ accounts for the additional refinery expansion capacity of each production unit $m \in M_{Ref}$ at refinery $i \in I$ for a specific expansion size $s \in S$. The integer variable $y_{exp}{}^{Ref}_{m,i,s}$ represents the decision of expanding a production unit and it can take a value of one if the unit expansion is required or zero otherwise. Constraint (10) sets an upper bound on intermediate streams flowrates between the different refineries. The integer variable $y_{pipe}{}^{Ref}_{cir,i,i'}$ represents the decision of exchanging intermediate products between the refineries and takes on the value of one if the commodity is transferred from plant $i \in I$ to plant $i' \in I$ or zero otherwise, where $i \neq i'$. When an intermediate stream is selected to be exchanged between two refineries, its flowrate must be below the transferring pipeline capacity $F^{U}_{cir,i,i'}$. Constraint (11) stipulates that the final products from each refinery $x^{Ref}_{cfr,i}$ less the amount exported $e^{Ref}_{cfr',i}$ for each exportable product $cfr' \in PEX$ from each plant $i \in I$ must satisfy the domestic demand $D_{Ref_{cfr}}$. The recourse variables $V^{Ref+}_{cfr,k}$, $V^{Ref-}_{cfr,k}$, $V^{Pet+}_{cp,k}$ and $V^{Pet-}_{cp,k}$ in equations (11) and (13) represent the refinery production shortfall and surplus as well as the petrochemical production shortfall and surplus, respectively, for each random realization $k \in N$. These variables will compensate for the violations in equations (11) and (13) and will be penalized in the objective function using appropriate shortfall and surplus costs C^{Ref+}_{cfr} and C^{Ref-}_{cfr} for the refinery products, and C^{Pet+}_{cp} and C^{Pet-}_{cp} for the petrochemical products, respectively. Resources are limited by constraint (12)

Constraints (13) and (14) represent the material balance that governs the operation of the petrochemical system. The petrochemical network receives its feed from potentially three main sources. These are, 1) refinery intermediate streams $Fi^{Pet}_{cr,cir,i}$ of an intermediate product $cir \in RPI$, 2) refinery final products $Ff^{Pet}_{cr,cfr,i}$ of a final product $cfr \in RPF$, and 3) non-refinery streams Fn^{Pet}_{cp} of a chemical $cp \in NRF$. For a given subset of chemicals $cp \in CP$, the proposed model selects the feed types, quantity and network configuration based on the final chemical and petrochemical lower and upper product demand $D_{Pet}{}^{L}_{cp}$ and $D_{Pet}{}^{U}_{cp}$ for each $cp \in CFP$, respectively. Furthermore, in equation (13) an additional term xi^{Pet}_{cp} was added to the left hand side representing the flow of intermediate petrochemical stream of $cp \in CIP$. In constraint (15), defining a binary variables $y_{proc}{}^{Pet}_{m}$ for each process $m \in M_{pet}$ is required for the process selection requirement as $y_{proc}{}^{Pet}_{m}$ will equal 1 only if process m is selected or zero otherwise. Furthermore, if only process m is selected, its production level must be at least equal to the process minimum economic capacity B^{L}_{m} for each $m \in M_{pet}$, where K^{U} is a valid upper bound. Finally, we can specify limitations on the supply of feedstock Fn^{Pet}_{cp} for each chemical type $cp \in NRF$ through constraint (18).

3. SCENARIO GENERATION

The solution of stochastic problems is generally very challenging as it involves numerical integration over the random continuous probability space of the second stage variables (Goyal & Ierapetritou, 2007). An alternative approach is the discretization of the random space using a finite number of scenarios. In our study, the Sample Average

Approximation (SAA) method, also known as stochastic counterpart, is employed. The SAA problem can be written as (Verweij et al., 2003):

$$v_N = \min_{x \in X} c^T x + \frac{1}{N} \sum_{k \in N} Q(x, \xi^k) \quad (19)$$

It approximates the expectation of the stochastic formulation (usually called the "true" problem) and can be solved using deterministic algorithms. Problem (19) can be solved iteratively in order to provide statistical bounds on the optimality gap of the objective function value. The validation procedure was originally suggested by Norkin et al. (1998) and further developed by Mark et al. (1999).

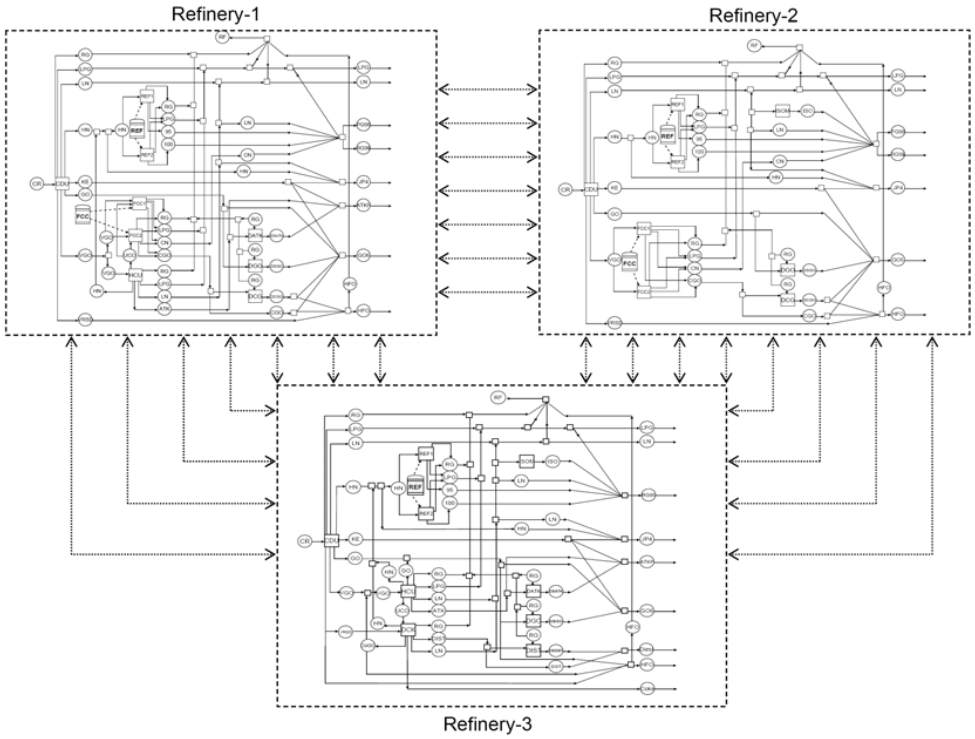


Fig. 1. Refinery Integration Network

originally suggested by Norkin et al. (1998) and further developed by Mark et al. (1999).

Table 1. Major refinery network capacity constraints

Production Capacity	Higher limit (10^3 ton/yr)		
	R1	R2	R3
Distillation	45000.	12000.0	9900.0
Reforming	700.0	2000.0	1800.0
Isomerization	200.0	-	450.0
Fluid catalytic cracker	800.0	1400.0	-
Hydrocracker	-	1800.0	2400.0
Delayed coker	-	-	1800
Des gas oil	1300.0	3000.0	2400.0
Des cycle gas oil	200.0	750.0	-
Des ATK	-	1200.0	1680.0
Des Distillates	-	-	450.0
Crude availability			
Arabian Light		31200.0	
Local Demand			
LPG		$\mathcal{N}$(432,20)	
LN		-	
PG98		$\mathcal{N}$(400,20)	
PG95		$\mathcal{N}$(4390,50)	
JP4		$\mathcal{N}$(2240,50)	
GO6		$\mathcal{N}$(4920,50)	
ATK		$\mathcal{N}$(1700,50)	
HFO		$\mathcal{N}$(200,20)	
Diesel		$\mathcal{N}$(400,20)	
Coke		$\mathcal{N}$(300,20)	

4. ILLUSTRATIVE CASE STUDY

This section presents the computational results of the proposed model and sampling scheme. The case study considers a subsystem of the petrochemical industry for the integration problem with the refinery network as apposed to considering the full scale petrochemical industry, which might have limited applications. The case study will examine the integration between a multirefinery network with a polyvinyl chloride (PVC) petrochemical complex. PVC is a major ethylene derivative with many important applications and uses (e.g. pipe fittings, automobile bumpers, toys, bottles, etc.).

In this paper, we consider the planning for three refineries in one industrial location, which is a common situation in many

areas around the world. The state equipment network (SEN) representation of the three refineries is shown in Fig. 1. The final products of the three refineries network consists of liquefied petroleum gas (LPG), light naphtha (LT), two grades of gasoline (PG98 and PG95), No. 4 jet fuel (JP4), military jet fuel (ATKP), No.6 gas oil (GO6), diesel fuel (Diesel), heating fuel oil (HFO), and petroleum coke (coke). The major capacity constraints for the refinery network are given in Table 1. The petrochemical complex, on the other hand, starts with the production of ethylene from the refineries feedstocks by steam cracking. The main feedstocks

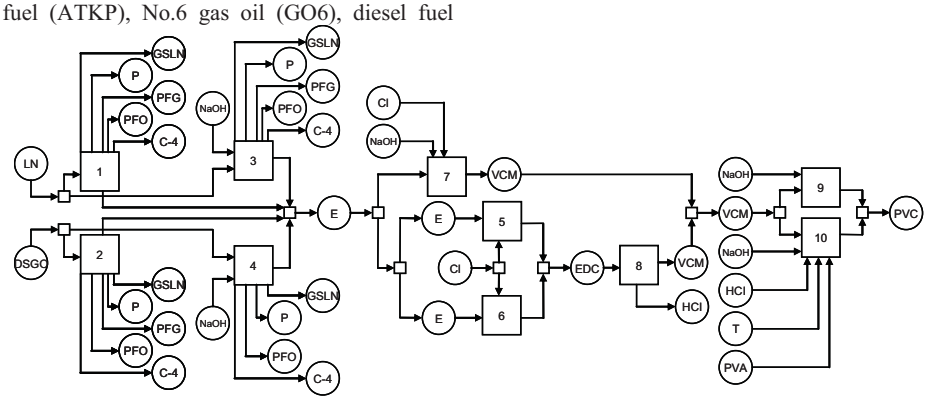


Fig. 2. PVC complex possible production alternatives

to the ethylene plant in our study are light naphtha (LN) and gas oil (GO). The selection of the feedstocks and hence the process technologies is decided upon based on the optimal balance and trade-off between the refinery and petrochemical markets. The process technologies considered in this study for the production of PVC are list in Table 2. The overall topology of all petrochemical technologies for the PVC production is shown in Fig. 2. The modeling system GAMS (Brooke et al., 1996) is used for setting up the optimization models and the MILP problems were solved with CPLEX (CPLEX Optimization Inc., 1993).

Table 2 Major products and processes in PVC complex

Product	Price ($/ton)	Process Technology	Index	Min. Prod. (10^3 ton/yr)
Ethylene (E)	$\mathcal{N}$(1570,10)	Pyrolysis of naphtha (LS)	1	250
		Pyrolysis of gas oil (LS)	2	250
		Steam cracking of naphtha (HS)	3	250
		Steam cracking of gas oil (HS)	4	250
Ethylene Dichloride (EDC)	$\mathcal{N}$(378,10)	Chlorination of ethylene	5	180
		Oxychlorination of ethylene	6	180
Vinyl chloride monomer (VCM)	$\mathcal{N}$(1230,10)	Chlorination and Oxychlorination of ethylene	7	250
		Dehydrochlorination of ethylene dichloride	8	125
Polyvinyl chloride (PVC)	$\mathcal{N}$(1600,10)	Bulk polymerization	9	50
		Suspension polymerization	10	90

In our study, we considered uncertainty in the imported crude oil price, refinery product price, petrochemical product price, refinery market demand, and petrochemical lower level product demand. In the presentation of the results, we focus on demonstrating the sample average approximation computational results as we vary the sample sizes and compare their solution accuracy and the CPU time required for solving the models.

Table 3 Computational results of stochastic model

		Lower bound sample size=N		
		1000	2000	3000
UB Samples	Number of Samples (R=30)			
N'=5000	LB estimate: $\overline{v}_N$	8802837	8804092	8804456
	LB error: $\widetilde{\varepsilon}_l$ (α=0.975)	3420	2423	1813
	UB estimate: $\hat{v}_{N'}$	8805915	8805279	8805578
	UB error: $\widetilde{\varepsilon}_u$ (α=0.975)	7776	7715	7778
	95% Conf. Interval	[0,14274]	[0,11324]	[0,10713]
	CPU (sec)	65	112	146
N'=10000	LB estimate: $\overline{v}_N$	8800071	8802080	8804305
	LB error: $\widetilde{\varepsilon}_l$ (α=0.975)	3356	2527	2010
	UB estimate: $\hat{v}_{N'}$	8803310	8803204	8803414
	UB error: $\widetilde{\varepsilon}_u$ (α=0.975)	5473	5833	5410
	95% Conf. Interval	[0,12068]	[0,9484]	[0,7420]
	CPU (sec)	196	224	263
N'=20000	LB estimate: $\overline{v}_N$	8796058	8801812	8802511
	LB error: $\widetilde{\varepsilon}_l$ (α=0.975)	3092	2345	1755
	UB estimate: $\hat{v}_{N'}$	8802099	8804121	8802032

UB error: $\widetilde{\varepsilon}_u$ (α=0.975)	3837	3886	3880
95% Conf. Interval	[0,12970]	[0,8540]	[0,5635]
CPU (sec)	1058	1070	1114

The problem was solved for different sample sizes N and N' to illustrate the variation of optimality gap confidence intervals, while fixing the number of replications R to 30. The replication number R need not be very large to get an insight of $\overline{v}_N$ variability. Table 4 shows different confidence interval values of the optimality gap when the sample size of N assumes values of 1000, 2000, and 3000 while varying N' between 5000, 10000, and 20000 samples. As the sample sizes N and N' were limited to these values due to computational considerations. In our case study, we ran into memory limitations when N and N' values exceeded 3000 and 20000, respectively. The solution of the three refineries network and the PVC complex using the SAA scheme with N = 3000 and N'=20000 required 1114 CPU sec to converge to the optimal solution.

Table 4. Model results integrated network

Process variables					Results (10^3 ton/yr) R1	R2	R3
Refinery	**Crude Oil Supply**				4500	12000	9900
	Production levels	Crude unit			4500	12000	9900
		Reformer			612.5	1824.6	1784.6
		Isomerization			160	-	450
		FCC			378	1174.2	-
		Hydrocracker			-	1740.4	2400
		Delayed coker			-	-	1440
		Des Gas oil			1300	3000	2400
		Des cycle gas oil			168.6	600	-
		Des ATK			-	1200	1654.8
		Des Distillates			-	-	366.2
	Intermediate streams exchange	From	**R1**	VGO	-	-	576.1 to HCU
			R2	LN	-	-	112.4 to Isom
			R3	VGO	-	274.8 to FCC	-
	Exports	PG95				439.8	
		JP4				1101.9	
		GO6				2044.2	
		HFO				1907.8	
		ATK				1887.6	
		Coke				110.7	
		Diesel				5.1	
Petrochemical	**Refinery feed to PVC complex**	Gas oil			788.6	1037.0	71.3
	Production levels	S. Crack GO (4)				486.8	
		Cl & OxyCl E (7)				475.4	
		Bulk polym. (9)				220.0	
	Final	PVC				220.0	
Total cost ($/yr)						$8,802,000	

Table 4 depicts the results of the optimal integration network between the three refineries and the PVC petrochemical complex. As shown in Table 5, the proposed model designed the refinery network and operating policies and also devised the optimal production plan for the PVC complex from all available process technologies. The model selected gas oil as the refinery feedstock to the petrochemical complex. PVC production was proposed by first high severity steam cracking of gas oil to produce ethylene. Vinyl chloride monomer (VCM) is then produced through the chlorination and oxychlorination of ethylene and finally, VCM is converted to PVC by bulk polymerization. The annual production cost across the refineries and the PVC complex was $8,802,000.

REFERENCES

Al-Qahtani, K., and Elkamel, A. (2008). Multisite Facility Network Integration Design and Coordination: An Application to the Refining Industry. *Computers & Chemical Engineering*, 32, 2198.

Al-Qahtani, K., Elkamel, A. and Ponnambalam, K. (2008). Robust Optimization for Petrochemical Network Design under Uncertainty, *Industrial & Engineering Chemistry Research*, 47, 3912.

Brooke, A., Kendrick, D., and Meeraus, A., Raman, R. (1996). *GAMS—A User's Guide*, GAMS Development Corporation: Washington DC.

CPLEX Optimization, Inc. (1993). *Using the CPLEX Callable Library and CPLEX Mixed Integer Library*, CPLEX Optimization Inc: Incline Village, NV.

Goyal, V., and Ierapetritou, M. G. (2007). Stochastic MINLP optimization using simplicial approximation. *Computers and Chemical Engineering*, 31, 1081.

Mark, W. K., Morton, D. P., and Wood, R. K. (1999). Monte Carlo bounding techniques for determining solution quality in stochastic programs. *Operational Research Letters*, 24, 47.

Norkin, W. I., Pflug, G. Ch., and Ruszczysk, A. (1998). A branch and bound method for stochastic global optimization. *Mathematical Programming*, 83, 425.

Verweij, B., Ahmed, S., Kleywegt, A. J., Nemhauser, G., and Shapiro, A. (2003). The sample average approximation method applied to stochastic routing problems: A computational study. *Computational Optimization & Applications*, 24, 289.

Nonlinear State Estimation of Differential Algebraic Systems

Ravi K. Mandela * **Raghunathan Rengaswamy** **
Shankar Narasimhan ***

* *Department of Chemical and Biomolecular Engineering, Clarkson University, Potsdam, NY 13699 USA (Tel: 315-268-3808; e-mail: mandelrk@clarkson.edu)*
** *Department of Chemical Engineering, Texas Tech University, Lubbock, TX 79409 USA (Tel: 806-742-3553; e-mail: raghu.rengasamy@ttu.edu)*
*** *Chemical Engineering Department, IIT Madras, India (e-mail:naras@iitm.ac.in)*

Abstract: Kalman filter and its variants have been used for state estimation of systems described by ordinary differential equation (ODE) models. Moving Horizon Estimation (MHE) has been a popular approach in chemical engineering community for the estimation of both ODE and differential algebraic equation (DAE) systems but is computationally demanding. There has been some work on applying Extended Kalman filter for state estimation of DAE systems with measurements as functions of only the differential states. This work describes the estimation of nonlinear DAE systems with measurements being a function of both the differential and algebraic states. An Unscented Kalman filter (UKF) formulation is also derived for semi-explicit index 1 DAE systems. The utility of these formulations are demonstrated through a case study.

1. INTRODUCTION

Differential algebraic equation (DAE) models naturally arise in several chemical/physical systems, where some rate processes are much faster than the others and admit quasi steady-state approximations. Common examples of these can be found in separation and reaction systems. Many chemical engineering systems can be modeled as DAE systems. Examples of algebraic equations include mole fraction summations, vapor-liquid equilibrium relationships and so on. The algebraic equations can be either linear or nonlinear. Other areas where DAE models arise are mechanical systems, electrical systems and biological systems. A DAE system is characterized by the index of the system. The index of a DAE system is defined as the number of differentiations that are required to convert the DAE system into an explicit ODE system. It is not always possible and easy to convert DAE into ODE systems [Petzold, 1988]. In this paper. the focus is on estimation of nonlinear index one DAE systems that are common in chemical engineering.

The Kalman filter (KF) is an optimal estimator for linear dynamical systems in the presence of state and measurement uncertainties [Gelb, 1988, Sorenson, 1985]. Extended Kalman filter (EKF) is an extension of the Kalman filter for nonlinear systems described by a class of ordinary differential equations. Simultaneous parameter and state estimation is achieved in KF and EKF by augmenting the states [Jazwinski, 1970].

The KF has been used by several researchers for state estimation of systems describing linear DAE models [Nikoukhah et al., 1992, Chisci and Zappa, 1992]. The state estimation of nonlinear DAEs has already been studied by Albuquerque and Biegler [1997] using Moving -horizon estimation technique. Moving-horizon estimation (MHE) is considered as an efficient optimization based method for state estimation. Moving-horizon estimation can also be extended to parameter estimation of nonlinear DAEs [Tjoa and Biegler, 1991].

Moving-horizon estimator can handle constraints and bounds at every sampling instant [Rao et al., 2003]. However, questions remain about the computational complexity for on-line implementation of MHE estimators. The main advantage of the EKF lies in their predictor-corrector recursive form that has the potential for online deployment [Muske and Edgar, 1997].

There has been some work on the application of EKF for nonlinear DAE systems. One of the first attempts at this can be found in Becerra et al. [1999]. Becerra et al. [2001] extend this work further and demonstrate their approach on an experimental case study. They also explore the use of square root formulation of the EKF which has better numerical stability than the standard EKF [Park and Kailath, 1995]. However, the measurements available to the estimator are all assumed to be functions of differential states. In this paper, we extend Becerra et al. [2001] approach to cases where the measurements are functions of both the differential and algebraic states. Further, we develop an approach for the use of Unscented Kalman filter (UKF) for estimation in index 1 nonlinear DAE systems.

The paper is organized as follows. Section 2 provides an introduction to DAE systems. EKF and UKF algorithms

for DAE systems are discussed in section 3 and section 4 respectively. Simulation results with discussions are presented in section 5 followed by conclusions in section 6.

2. DIFFERENTIAL ALGEBRAIC SYSTEMS

As discussed in the previous section, DAE systems consist of both differential and algebraic equations. DAE systems are characterized by the index of the system. The index of the DAE system is defined as the number of differentiations required to convert the DAE into an ODE. As a simple example, consider

$$\dot{y_2}(t) = y_1(t) + \lambda_1(t) \tag{1}$$

$$0 = y_2(t) + \lambda_2(t) \tag{2}$$

Differentiating the algebraic equation 2 once, we get

$$0 = \dot{y_2}(t) + \dot{\lambda_2}(t) \tag{3}$$

Differentiating the algebraic equation 3 once more yields

$$0 = \ddot{y_2}(t) + \ddot{\lambda_2}(t) \tag{4}$$

Putting these equations together we now get an ODE as shown in equation .

$$\begin{aligned} \dot{y_2}(t) &= y_1(t) + \lambda_1(t) \\ \dot{y_1}(t) &= -\dot{\lambda_1}(t) - \ddot{\lambda_2}(t) \end{aligned} \tag{5}$$

Since the equations had to be differentiated twice this is an index 2 DAE system. While there are DAE systems of orders higher than 1 in chemical engineering, index 1 DAE systems are common as seen in electrochemistry, reactive distillation and biochemical engineering applications. As mentioned before, this work considers index 1 DAE systems.

3. EKF FOR DAE SYSTEMS

While EKF has been studied extensively for ODE systems, the application of EKF approaches to DAE systems are not many. Becerra et al. [2001] developed an EKF estimation approach for for nonlinear index 1 DAEs. The EKF approach follows the same predictor-corrector form with some modifications. In the prediction step, a DAE solver is used for propagating the prior state through the system model. This is in contrast to the use of an ODE solver in standard EKF. The covariance matrix of the differential states are propagated by linearizing the system model. The correction step is performed only for the differential states through a linearization of the measurement model. This is possible because it is assumed that the measurements are functions of differential states alone. Once the corrected differential states are available, the corrected algebraic states are calculated using the algebraic portion of the system model. The corrected covariance matrix for the differential states is calculated using the standard EKF procedure. The mathematical details of the algorithm are explained below. The nonlinear DAE system is considered with discrete measurements sampled at regular intervals with sampling period Δt

$$x_{k+1} = x_k + \int_{(k)\Delta t}^{(k+1)\Delta t} f(x(\tau)), z(\tau))\, d\tau + w_{k+1} \tag{6}$$

$$g(x_{k+1}, z_{k+1}) = 0 \tag{7}$$

$$y_{k+1} = h(x_{k+1}) + v_{k+1} \tag{8}$$

where w_{k+1} and v_{k+1} are assumed to be independent Gaussian white noise processes with known covariance matrix Q_{k+1} and R_{k+1}

For a fixed input, the linearized equation is given by

$$\dot{x} = Ax \tag{9}$$

where

$$A = (J_1 - J_2 J_4^{-1} J_3) \tag{10}$$

$$\begin{bmatrix} J1 & J2 \\ J3 & J4 \end{bmatrix} = \begin{bmatrix} \frac{\partial f}{\partial x} & \frac{\partial f}{\partial z} \\ \frac{\partial g}{\partial x} & \frac{\partial g}{\partial z} \end{bmatrix} \tag{11}$$

Following are the steps involved in the algorithm

- The differential states are propagated by integrating the DAE model from time t_k to t_{k+1}. The predicted state estimate $\hat{x}_{k+1/k}$ is obtained with u_k, which is the constant input between sampling intervals.
- The predicted covariance matrix in differential states is propagated using
$$P_{k+1/k} = \bar{A}_k P_{k/k} \bar{A}_k^T + Q_k \tag{12}$$
where $\bar{A} = exp(A\Delta t)$
- The kalman gain is computed using
$$K_{k+1} = P_{k+1/k} G_{k+1}^T (G_{k+1} P_{k+1/k} G_{k+1}^T + R_{k+1})^{-1} \tag{13}$$
where G_{k+1} is the linearized measurement model and the actual measurement model is a function of only differential states.
- The updated differential estimates are obtained from kalman update equation
$$\hat{x}_{k+1/k+1} = \hat{x}_{k+1/k} + K_{k+1}(y_{meas} - h(\hat{x}_{k+1/k})) \tag{14}$$
- The updated estimate $\hat{z}_{k+1/k+1}$ is obtained from the set of algebraic equations defining the DAE system once differential state estimate $\hat{x}_{k+1/k+1}$ is obtained
- The updated covariance matrix is computed as
$$P_{k+1/k+1} = (I - K_{k+1} G_{k+1}) P_{k+1/k} \tag{15}$$
In this method, $\hat{z}$ is computed only from the $\hat{x}$ using algebraic equation and there is no dependence or use of prior estimates of z (algebraic states). This method cannot be applied to cases where there is an availability of algebraic states measurements.

4. PROPOSED APPROACH: EXTENDED KALMAN FILTER FOR DAE SYSTEMS

In DAE systems, the measurements can, in general, be a function of both the differential and algebraic states. In the proposed work, we extend the EKF approach to this case. The algorithm deviates from the work of Becerra et al. [2001] in that the EKF works with an augmented system (with both the differential and algebraic states). A linearized ODE model involving both differential and

algebraic states (augmented) is derived. This linearized ODE model is used for the covariance propagation of augmented state as opposed to just the differential states as in Becerra et al. [2001]. The gain matrix is calculated form the augmented predicted covariance matrix and the linearized measurement model which is a function of both the differential and algebraic measurements. The corrected augmented state is computed. From these corrected augmented states, only the differential states are retained. As the algebraic constraints are to be met, the algebraic states are calculated from the corrected differential states using algebraic equations. The details of the algorithm are explained below. The nonlinear DAE system is considered with discrete measurements sampled at regular intervals with sampling period Δt

$$x_{k+1} = x_k + \int_{(k)\Delta t}^{(k+1)\Delta t} f(x(\tau)), z(\tau)) \, d\tau + w_{k+1} \tag{16}$$

$$g(x_{k+1}, z_{k+1}) = 0 \tag{17}$$

$$y_{k+1} = h(x_{k+1}) + v_{k+1} \tag{18}$$

where w_{k+1} and v_{k+1} are assumed to be independent Gaussian white noise processes with known covariance matrix Q_{k+1} and R_{k+1}

Linearizing the differential equations and algebraic equations of index 1 DAE system, we get

$$\begin{aligned} \dot{x} &= Ax + Bz \\ 0 &= Cx + Dz \end{aligned} \tag{19}$$

where

$$\begin{bmatrix} A & B \\ C & D \end{bmatrix} = \begin{bmatrix} \frac{\partial f}{\partial x} & \frac{\partial f}{\partial z} \\ \frac{\partial g}{\partial x} & \frac{\partial g}{\partial z} \end{bmatrix} \tag{20}$$

Differentiating the linearized algebraic equation once , we get

$$0 = C\dot{x} + D\dot{z} \tag{21}$$

Then

$$\dot{z} = -D^{-1}C\dot{x} \tag{22}$$

$$\dot{z} = -D^{-1}CAx - D^{-1}CBx \tag{23}$$

Writing in matrix form

$$\begin{bmatrix} \dot{x} \\ \dot{z} \end{bmatrix} = \begin{bmatrix} A & B \\ -D^{-1}CA & -D^{-1}CB \end{bmatrix} \begin{bmatrix} x \\ z \end{bmatrix} \tag{24}$$

The augmented form is

$$\dot{X}^{aug} = A^{aug} X^{aug} \tag{25}$$

The transition matrix is evaluated as

$$\phi = exp(A^{aug} \Delta t) \tag{26}$$

The algorithm consists of following steps

- Both differential and algebraic states are propagated using a DAE solver from t_k to t_{k+1} starting from the latest updated estimate $\hat{X}_k^{aug}$ and the latest input u_k.
- The predicted covariance matrix of the augmented states is computed as
$$P_{k+1/k}^{aug} = \phi P_{k/k}^{aug} \phi^T + \Gamma Q_{k+1} \Gamma^T \tag{27}$$
where
$$\Gamma = \begin{bmatrix} I \\ -D^{-1}C \end{bmatrix} \tag{28}$$
- The augmented Kalman gain is computed as
$$K_{k+1}^{aug} = P_{k+1/k}^{aug} G_{k+1}^T (G_{k+1} P_{k+1/k}^{aug} G_{k+1}^T + R_{k+1})^{-1} \tag{29}$$
where G_{k+1} is the linearised measurement model.
- The updated state estimate is given by
$$X_{k+1/k+1}^{aug} = X_{k+1/k}^{aug} + K_{k+1}^{aug} (y_{meas} - h(X_{k+1/k}^{aug}) \tag{30}$$
- As the algebraic constraints are to be met, differential terms (x) of the updated estimate are retained and the updated estimates of the algebraic states (z) are calculated from the algebraic equation of DAE system.
- The updated covariance matrix is calculated as
$$P_{k+1/k+1}^{aug} = (I - K_{k+1}^{aug} G_{k+1}) P_{k+1/k}^{aug} \tag{31}$$

5. UNSCENTED KALMAN FILTER FOR DAE SYSTEMS

Unscented Kalman filter (UKF) is an approach that was developed to improve on EKF. The UKF approach uses the idea of unscented transforms for predicting the mean and covariance when a random variable passes through a nonlinear transformation. In EKF, linearization of the nonlinear transformation is used to predict the mean and covariance of the transformed variable. Unscented transformation is a sampling technique where a small number of deterministic samples are chosen such that their weighted mean and covariance exactly equal the mean and covariance of the random variable undergoing the nonlinear transformation. The transformed sample points are used to calculate the *a posteriori* mean and covariance. This results in much better accuracy than the linearization approach [Julier et al., 2000].

UKF estimation for ODE systems is well developed and several application studies have appeared [Romanenko and Castro, 2004, Romanenko et al., 2004, van der Merwe et al., 2000, Julier, 2002, Wan et al., 2000, Wan and van der Merwe, 2000]. In this paper, we extend the UKF approach for semi-explicit index 1 DAE systems. The proposed approach also follows the predictor-corrector form. First, unscented samples are chosen for the differential states. The unscented samples for the algebraic states are generated from the algebraic equations. This makes all the sigma points consistent. These sigma points are propagated through the system through a DAE solver. Unscented samples for the differential and algebraic states are again generated using the propagated covariance matrix. The sample points for the measurements are calculated by passing the unscented differential and algebraic state samples through the measurement function. The sample covariances are used to calculate the Kalman gain. Using the Kalman gain, the corrected differential states are obtained. The corrected algebraic states are calculated using the algebraic equations in the system model. This algorithm of unscented Kalman filter for DAE systems is

explained below. The nonlinear DAE system is considered with discrete measurements sampled at regular intervals with sampling period Δt

$$x_{k+1} = x_k + \int_{(k)\Delta t}^{(k+1)\Delta t} f(x(\tau)), z(\tau))\, d\tau + w_{k+1} \quad (32)$$

$$g(x_{k+1}, z_{k+1}) = 0 \quad (33)$$

$$y_{k+1} = h(x_{k+1}, z_{k+1}) + v_{k+1} \quad (34)$$

where w_{k+1} and v_{k+1} are assumed to be independent Gaussian white noise processes with known covariance matrix Q_{k+1} and R_{k+1}

- The first step is the generation of sigma points. At the k^{th} instant, $\hat{x}_{k/k}$ is the filtered estimate of differential states and $P_{k/k}$ is the covariance matrix associated with it. A set of 2n+1 sigma points $\hat{X}_{k/k,i}$ with associated weights are chosen symmetrically about $\hat{x}_{k/k}$ where n is the dimension of the state.

$$\hat{X}_{k/k,0} = \hat{x}_{k/k}; W_0 = \frac{\kappa}{(n+\kappa)} \quad (35)$$

$$\hat{X}_{k/k,i} = \hat{x}_{k/k} + (\sqrt{(n+\kappa)P_{k/k}})_i; W_i = \frac{1}{2(n+\kappa)} \quad (36)$$

$$\hat{X}_{k/k,i+n} = \hat{x}_{k/k} - (\sqrt{(n+\kappa)P_{k/k}})_i; W_{i+n} = \frac{1}{2(n+\kappa)} \quad (37)$$

where $(\sqrt{P_{k/k}})_i$ is the i^{th} column of matrix square root of $P_{k/k}$ and W_i is the weight associated with the corresponding point. The parameter κ is a tuning parameter. The weights W_i add to one and the weighted mean of the set X is same as $\hat{x}_{k/k}$. The weighted covariance matrix of the sample is equal to $P_{k/k}$.

$$P_{k/k} = \sum_{i=0}^{2n} W_i (\hat{X}_{k/k,i} - \hat{x}_{k/k})(\hat{X}_{k/k,i} - \hat{x}_{k/k})^T \quad (38)$$

- Calculate $\hat{Z}_{k/k,i}$ from $g(\hat{X}_{k/k,i}, \hat{Z}_{k/k,i}) = 0$
- Propagate $\hat{X}_{k/k,i}$ and $\hat{Z}_{k/k,i}$ through DAE system to get $\hat{X}_{k+1/k,i}$ and $\hat{Z}_{k+1/k,i}$

  The predicted differential state estimate $\hat{x}_{k+1/k}$ is given by

$$\hat{x}_{k+1/k} = \sum_{i=0}^{2n} W_i \hat{X}_{k+1/k,i} \quad (39)$$

- Calculate $P^{xx}_{k+1/k}$

$$P^{xx}_{k+1/k} = \sum_{i=0}^{2n} W_i (\hat{X}_{k+1/k,i} - \hat{x}_{k+1/k}) (\hat{X}_{k+1/k,i} - \hat{x}_{k+1/k})^T + Q_{k+1} \quad (40)$$

- Do unscented sampling with $\hat{x}_{k+1/k}$ as mean and $P^{xx}_{k+1/k}$ as covariance matrix
- Recalculate $\hat{Z}_{k+1/k,i}$ from $g(\hat{X}_{k+1/k,i}, \hat{Z}_{k+1/k,i}) = 0$
- Form $\hat{X}^{aug}_{k+1/k,i}$ by augmenting $\hat{X}_{k+1/k,i}$ with $\hat{Z}_{k+1/k,i}$
- Calculate $\hat{x}^{aug}_{k+1/k}$

$$\hat{x}^{aug}_{k+1/k} = \sum_{i=0}^{2n} W_i \hat{X}^{aug}_{k+1/k,i} \quad (41)$$

- The predicted sigma points are propagated through the nonlinear measurement equation to obtain the predicted measurement as

$$Y_{k+1,i} = h(\hat{X}^{aug}_{k+1/k,i}) \quad (42)$$

Using the set of predicted measurements, the covariance matrix of innovations and the cross covariance between predicted state estimate errors and innovations are computed as

$$P_{\nu\nu,k+1} = \sum_{i=0}^{2n} W_i (Y_{k+1,i} - \hat{y}_{k+1}) (Y_{k+1,i} - \hat{y}_{k+1})^T + R_{k+1} \quad (43)$$

$$P_{x\nu,k+1} = \sum_{i=0}^{2n} W_i (\hat{X}^{aug}_{k+1,i} - \hat{x}^{aug}_{k+1/k})(Y_{k+1,i} - \hat{y}_{k+1})^T \quad (44)$$

where

$$\hat{y}_{k+1} = \sum_{i=0}^{2n} W_i Y_{k+1,i} \quad (45)$$

- The Kalman gain matrix is computed as

$$K_{k+1} = P_{x\nu,k+1} (P_{\nu\nu,k+1})^{-1} \quad (46)$$

- The Kalman gain corresponding to differential states is K^{diff}_{k+1}
- The updated differential estimates are obtained using the linear update equation as in Kalman filter

$$\hat{x}_{k+1/k+1} = \hat{x}_{k+1/k} + K^{diff}_{k+1} (y_{k+1} - \hat{y}_{k+1}) \quad (47)$$

- The updated estimate $\hat{z}_{k+1/k+1}$is obtained from the set of algebraic equations defining the DAE system once differential state $\hat{x}_{k+1/k+1}$ is obtained
- The covariance matrix of error in the updated differential estimates is computed using

$$P_{k+1/k+1} = P_{k+1/k} - K^{diff}_{k+1} P_{\nu\nu,k+1} K^{diff\,T}_{k+1} \quad (48)$$

6. CASE STUDY

The utility of the proposed approaches is tested on an electrochemical case study. The case study considers the galvanostatic charge /open-circuit/ discharge processes of a thin film nickel hydroxide electrode [Celik et al., 2002]. The modeling equations are

$$\frac{\rho V}{W} \frac{dy_1}{dt} = \frac{j_1}{F} \quad (49)$$

$$j_1 + j_2 - i_{app} = 0 \quad (50)$$

where

$$j_1 = i_{01}[2(1-y_1) exp(\frac{0.5F}{RT}(y_2 - \phi_{eq,1})) - 2y_1 \times exp(\frac{-0.5F}{RT}(y_2 - \phi_{eq,1}))] \quad (51)$$

$$j_2 = i_{02}[exp(\frac{F}{RT}(y_2 - \phi_{eq,2})) - exp(\frac{-F}{RT}(y_2 - \phi_{eq,2}))] \quad (52)$$

The first equation is the species balance equation, the second equation is the charge balance equation and j_1 and j_2 are derived using the Butler-Volmer kinetics. For the purpose of demonstrating the utility of the proposed approaches we assume that the differential state is corrupted with process noise w_{k+1} and the algebraic equation is exact. The values of parameters used are $F = 96487$, $R = 8.314$, $T = 298.15$, $\phi_{eq,1} = 0.420$, $\phi_{eq,2} = 0.303$, $\rho = 3.4$, $W = 92.7$, $V = 1 \times 10^{-5}$, $i_{app} = 1 \times 10^{-5}$, $i_{01} = 1 \times 10^{-04}$, $i_{02} = 1 \times 10^{-08}$. The units of parameters and variables are omitted for the simplicity. y_1 is the mole fraction of Nickel hydroxide and y_2 is potential difference between at the solid-liquid interface.The initial guess to the estimator is $[x0, z0] = [0.5322, 0.4254]$ and the actual value is [0.35024, 0.4071]. The tuning parameters used in EKF are

The following parameters are used

$$\Delta t = 15$$
$$P_0 = \begin{bmatrix} 0.005 & 0 \\ 0 & 0.005 \end{bmatrix}$$
$$Q_{k+1} = 0.00001$$
$$R_{k+1} = 0.0001$$

where Δt is the sampling time , P_0 is the error covariance matrix of differential and algebraic states, Q_{k+1} is the process noise associated with differential states and R_{k+1} is the measurement covariance matrix. The measurement in this case study is y_2, which is the potential difference at the solid-liquid interface. The important point to note is that the augmented covariance matrix should be taken into consideration if the measurement model is a function of differential and algebraic states. Figure 1 and Figure 2 show the estimates for the mole fraction and potential difference.

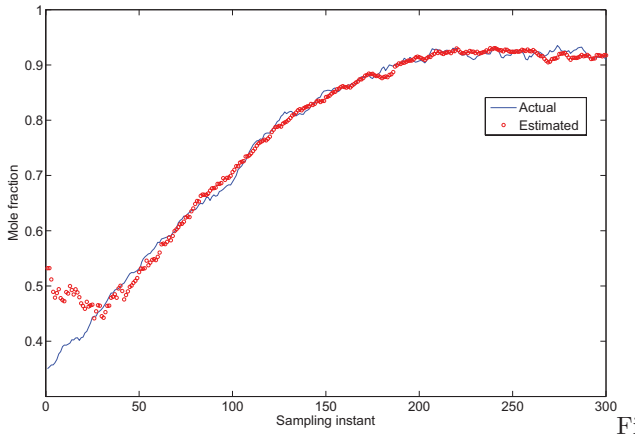


Fig. 1. EKF estimates of mole fraction for case study

The same differential algebraic system is considered and the UKF approach proposed in this paper is tested. The main advantage of UKF lies in the fact that it does not require linearization to compute covariance matrices. The UKF estimator gives very good estimates of mole fraction and potential difference as shown in Figure 3 and Figure

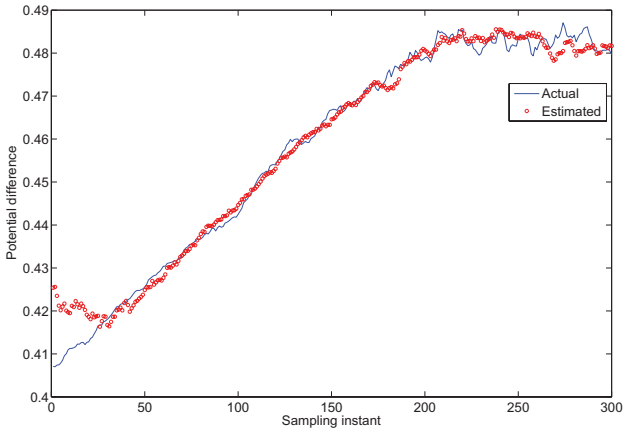


Fig. 2. EKF estimates of potential difference for case study

4. The tuning parameters for the UKF are same as used in EKF implementation. Figure 5 shows the comparison of UKF and EKF estimates and the their performances are compared by computing the root mean square error (RMSE) of the two states. Table 6 shows the RMSE values of estimates of UKF and EKF. It can clearly be seen that the UKF performs better than the EKF for this case study. Further, the UKF also avoids linearization in the computation of the covariance matrices.

RMSE values of EKF and UKF		
Method	RMSE y_1	RMSE y_2
EKF	0.0305	0.0035
UKF	0.0035	0.0035

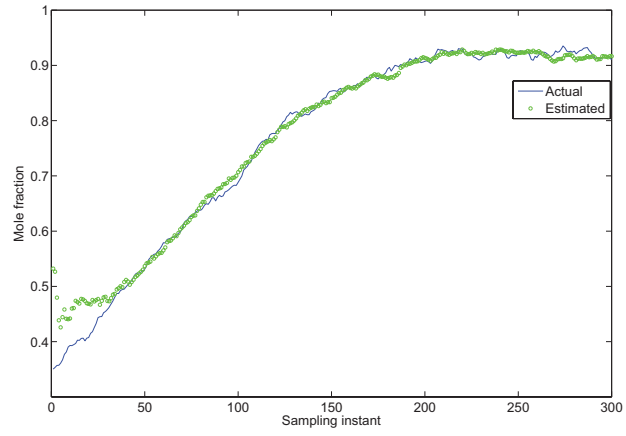


Fig. 3. UKF estimates of mole fraction for case study

7. CONCLUSIONS

In this paper, EKF and UKF formulations for nonlinear DAEs were proposed. The proposed EKF approach handles the case where the measurement functions are a function of both the differential and algebraic states. While UKF for ODE systems are well studied, there is very little work on the application of the UKF approach to DAE

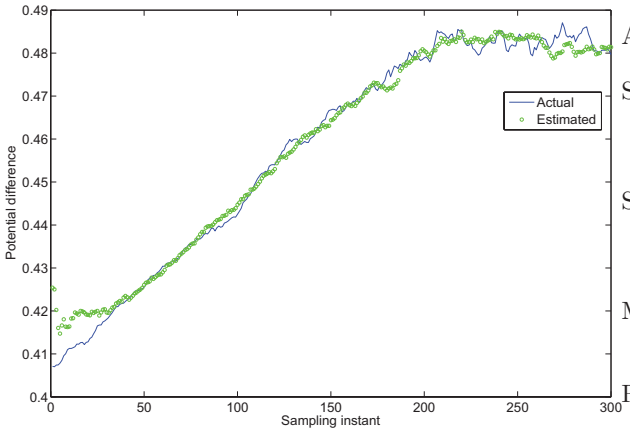


Fig. 4. UKF estimates of potential difference for case study

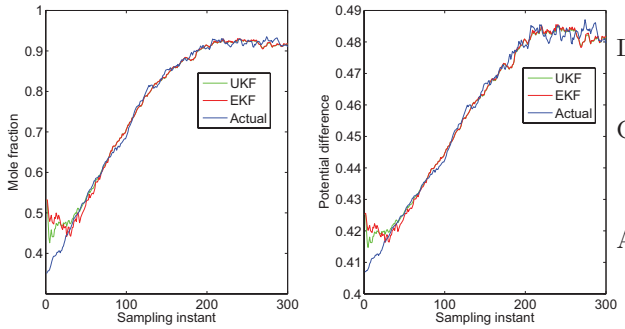


Fig. 5. Comparison of UKF and EKF estimates

systems. One possible approach to use unscented transformation in the estimation of DAE systems is proposed in this work. A case study is presented to demonstrate both the approaches. In this case study, the algebraic state is directly measured. It is shown that while both the proposed approaches provide satisfactory estimation, the UKF approach outperforms the EKF approach.

REFERENCES

J. S. Albuquerque and L. T. Biegler. Decomposition algorithms for on-line estimation with nonlinear models. *Computers in Chemical Engineering*, 21:283–299, 1997.

V. M. Becerra, P. D. Roberts, and G. W. Griffiths. Dynamic data reconciliation for a class of nonlinear differential algebraic equation models using the extended kalman filter. *Proceedings of the 14th IFAC world congress*, pages 303–308, 1999.

V. M. Becerra, P. D. Roberts, and G. W. Griffiths. Applying the extended kalman filter to sytems described by nonlinear differential-algebraic equations. *Control Engineering Practice*, 9:267–281, 2001.

E. Celik, E. Karaduman, and B. Mustafa. Numerical method to solve chemical differential algebraic equations. *International journal of quantum chemistry*, 89: 447–451, 2002.

L. Chisci and G. Zappa. Square root kalman filtering of descriptor systems. *Systems and Control Letters*, 19(4): 325–334, 1992.

A. Gelb. *Applied optimal estimation*. MIT Press, Cambridge, 1988.

A. Jazwinski. *Stochastic processes and filtering theory*. Academic Press, New York, 1970.

S. Julier, J. Uhlmann, and H. Whyte. A new method for the nonlinear transformation of means and covaraiances in filters and estimators. *IEEE Trans. Automat. Control*, 45:477–482, 2000.

S. J. Julier. The scaled unscented transformation. In *Proceedings of American Control Conference 2002*, volume 6, pages 4555–4559. American Control Conference, 2002.

M. Muske and T. Edgar. *Nonlinear state estimation. In: Henson, M.A, Seborg D.E, eds. Nonlinear Process Control*. Prentice Hall, Upper Saddle River, NJ, 1997.

R. Nikoukhah, A. Willsky, and B. C. Levy. Kalman filltering and riccati equations for descriptor systems. *IEEE Transactions on Automatic Control*, 37(9):1325–1342, 1992.

P. Park and T. Kailath. New square root algorithms for kalman filtering. *IEEE Transactions on Automatic Control*, 40(5):895–899, 1995.

L. R. Petzold. Differerential algebraic equations are not ode's. *SIAM Journal of Science and Statistical Computing*, 3:367–384, 1988.

C. Rao, J. Rawlings, and D. Mayne. Constrained state estimation for nonlinear discrete-time systems: stability and moving horizon approximations. *IEEE Trans. Automat. Control*, 48:246–258, 2003.

A. Romanenko and J. A. A. M. Castro. The unscented filter as an alternative to the ekf for nonlinear state estimation: A simulation case study. *Comput. Chem. Eng.*, 28(3):347–355, 2004.

A. Romanenko, L. O. Santos, and P. A. F. N. A. Afonso. Unscented kalman filtering of a simulated ph system. *Ind. Eng. Chem. Res.*, 43(23):7531–7538, 2004.

H. Sorenson. *Kalman filtering theory and applications*. IEEE Press, New York, 1985.

I. B. Tjoa and L. T. Biegler. Simultaneous solution and optimization strategies for parameter estimation of differential-algebraic equation systems. *Industrial and Engineering Chemistry Research*, 30(2), 1991.

R. van der Merwe, A. Doucet, J. F. G. de Freitas, and E. A. Wan. The unscented particle filter. Technical report, Cambridge University Engineering Department, 2000.

E. A. Wan and R. van der Merwe. The unscented kalman filter for nonlinear estimation. In *Proceedings of IEEE Symposium 2000(AS-SPCC)*, Alberta, Canada, October 2000.

E. A. Wan, R. van der Merwe, and A. T. Nelson. Dual estimation and the unscented transformation. *Neural Information Processing Systems*, 12:666–672, 2000.

RIVER WATER QUALITY MODEL VERIFICATION THROUGH A GIS BASED SOFTWARE

M. K. Yetik*, M. Yüceer, R. Berber***, E. Karadurmuş******

** Turkish Statistical Institute Regional Office, Zonguldak, Turkey, (e-mail: kazim.yetik@tuik.gov.tr).*
*** Department of Chemical Engineering, Faculty of Engineering Inonu University, 44280 Malatya, Turkey, (e-mail:myuceer@inonu.edu.tr)*
**** Department of Chemical Engineering, Faculty of Engineering Ankara University, Tandoğan 06100 Ankara, Turkey (e-mail: berber@eng.ankara.edu.tr)*
***** Department of Chemical Engineering, Faculty of Engineering Hitit University, Çorum, Turkey (e-mail: erdalk@gazi.edu.tr)*

Abstract: Research and development attempts on water quality models created valuable resources in the sense of model calibration and verification techniques. Recognizing the current degree of pollution in rivers and the importance of the sustainable water resources management, the interactive river monitoring appears to be at the center of recent focus. However the available information in this area is still far from expectations. On one side, the Geographical Information Systems (GIS) are gaining widespread acceptance and on the other side fast and reliable water quality models and parameter estimation techniques are becoming available. However, previous work on integrating water quality models and GIS is very limited. This work brings an integrated platform on which ArcMap as a GIS and a water quality model in Matlab™ are brought together in an interactive and user-friendly manner. The software developed allows the user to enter the data collected from the river, runs the dynamic model in the Matlab™ environment, predicts the values of pollution constituents along the river, extracts the results and displays the water quality on the map in different forms. The software thus provides a considerable ease in future real time application for on site river monitoring and environmental pollution assessment.

Keywords: water quality modeling, GIS, GUI, water management, systems analysis.

1. INTRODUCTION

Water pollution is gradually becoming one of major threats for aquatic as well as human life. In order to assess the impact of wastewater discharges into the surface waters, mathematical models are of great importance. Over the past there have been considerable developments in the area of water quality modeling for rivers. A summary can be found in the review by Rauch *et al.* (1998) who gave the then state of the art in river water quality modeling. The most widely used model in the world is pronounced to be QUAL2E, which was developed by US Environmental Protection Agency (EPA), and known as almost the standard for river water quality modeling (Shanahan *et al.* 1998). In addition; WASP, SALMANQ and SIMCAT are probably the ones that have been frequently referred to in the literature. The water quality models can be classified from many perspectives, ranging from model complexity to the simulation method employed, and the number and type of water quality indicators incorporated. Just to give an idea, Cox (2003), for example, selected 6 models in conceptualization and solution for detailed comparison. Three of them were steady state and the rest was of dynamic character. Cox (2003) noted that water quality modeling was an active area of research around the world, and underlined that only few papers referred to specific models with majority of the papers reporting applications with QUAL2E.

In the authors' research group, a dynamic modeling strategy based on QUAL2E and coupled with a parameter estimation technique was introduced by Karadurmus and Berber (2004). The suggested strategy assumed that river reach could be modeled as a single CSTR. The model predicted and compared to the field data for 10 quality constituents observed; except those for the total coliform, total chloride and BOD_5, good agreement was obtained. Later a user-interactive software code in Matlab™ (The MathWorks Inc., USA) named as RSDS (River Stream Dynamics and Simulation) for the implementation of the suggested technique was presented, and the model predictions were compared against experimental data collected in field observations along the Yesilirmak river basin in Turkey and predictions from QUAL2E (Yuceer *et al.* 2007). In a following work, a water reach was represented by a series of CSTRs rather than a single one. Taking the trade-off between the computing load and the prediction accuracy into account, the number of CSTRs to be used to represent a river section was determined. Then, for simulating a 500 m long reach of the river between the two sampling stations, 20 CSTRs were used (Berber *et al.* 2009). Furthermore, this work included a parameter identification study.

Despite the progress that has been observed in the field of modeling, only few reports are available in the current literature on integrated software development for river water quality monitoring. It is seen that the recent efforts are now concentrating on the incorporation of a geographical information system to water quality models. Within this framework, Marsili–Libelli *et al.* (2001) described the interfacing of a Matlab™ based quality model to a popular geographical information system ArcView™ (ESRI Inc., USA) by a communication protocol through which data could be exchanged between the two platforms. The same research group later provided a new software package developed entirely in the Matlab™ platform based on the Mapping Toolbox™ and reported enhanced interactivity and portability. The features of the program are illustrated through a case study (Marsili–Libelli *et al.* 2002).

From the perspective of using web-based technologies for remote monitoring, Cianchi *et al.* (2000) used internet technologies to follow water quality with river quality sensors. Data from sensor signals were transmitted to information warehouse by internet. In a more recent work, web based Geological Information System was used to visualize and assess water quality over the web for end user with minimum knowledge and computing experience (Ganapathy and Ernest, 2004). The spatial 'Decision Support System' developed for their study focused on the lower Rio Grand river basin.

The use of Geographical Information System (GIS) computing platforms, as they represent a process for looking at geographic patterns in data and provide nice display options, has been increasing. GIS incorporates computer hardware, software, and geographic data for capturing, managing, analyzing, and displaying all forms of geographically referenced information. This rapidly growing technological field brings graphical features with tabular data in order to assess real-world problems. The opportunities that GIS systems provide may range from simple applications where one layer data display and analysis is done on a digital map, to more complex cases that mimic the real world by combining many data layers (Mitchell, 1999). Distributions of nitrate, nitrite and ammonium at various monitoring sites across the Humber basin were examined by Davies and Neal (2004) within a GIS framework. Empirical relationships between land characteristics and water quality for the whole catchment draining to each water quality monitoring site were established. The main water quality data source was the Land Ocean Interaction Study dataset. The land characteristics were classfied as lowland arable, urban, upland and coniferous woodland. The relationship between water quality and the catchment characteristics were assessed using linear regression. The study has proved success in showing the broad patterns across the region based on regression analysis of environmental measurements on the nitrogen species and simple land characteristics. (Davies and Neal, 2004). In a particular work by Ruelland *et al.* (2007) the Riverstrahler model that describes the biological functioning of an entire river system was coupled to a GIS interface to make the model entirely generic to be run on any river system for which a suitable database was available. They examined the effect of increasing the spatial resolution of the drainage network representation on the performance of the Riverstrahler model.

In this study we have developed an interactive GIS based software for water quality monitoring in rivers. The water quality model that has been previously developed in our research group was used for simulation and prediction. The software created has been tested with off-line water quality data gathered from a 36.5 km long section of Yesilirmak river in the central northern region of Turkey.

2. GIS PLATFORM INTEGRATING A WATER QUALITY MODEL IN MATLAB

A software has been created in this work to analyze the river water quality data in GIS platform. The program, called RSDS-C, and particularly designed to simulate Yesilirmak river in the central northern part of Turkey, allows user interaction and visual effects so that the predictions for pollution constituents can be represented on a digital map of the river. The River Stream Dynamic Simulation (RSDS) software previously developed in Matlab™ in our research group (Yuceer *et al.* 2007) was used as the water quality model, and was incorporated into the GIS platform ArcMap™ 9.1.

One critical point in combining a Matlab model with a GIS system is integrating the geographical data (which come from digital maps of GIS) with river pollution variables that are handled in Matlab. Data exchange between these two platforms requires that the graphical indications used to represent the geographical object in GIS be adapted to the data structure in the model embodied in Matlab. We used the data transfer strategy depicted in Fig. 1, which shows that the ASCII formatted text files were the medium of transfer between GIS (ArcMap™) and Matlab™. As ArcMap™ employs database files for displaying the digital maps, Microsoft Access™ was employed as the database-handling platform.

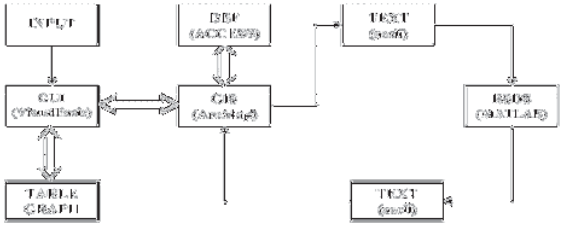


Fig. 1. ASCII file transfer strategy between different computing platforms.

A special graphical user interface (GUI) was designed for data input related to the river. The input comprise initial conditions for simulation, parameters related to integration of differential equations, flow characteristics of the river, or real measured data that has been observed at a particular location along the river (particularly when a parameter estimation study is intended). The GUI allows the user to interactively enter the observed quality of the river, which may be used as

the initial conditions at the beginning of simulations, or as the experimental value for the embedded simulation algorithm in case if parameter identification is to be performed. As for the water quality constituents, we use 11 variables comprising dissolved oxygen, carbonaceous BOD, four nitrogen forms (organic, ammonia, nitrite, and nitrate), two phosphorus forms (organic and dissolved), coliforms, nonconservative constituent chloride and phytoplanktonic algae. Those are the state variables of the embedded rigorous water quality model (Yuceer *et al.* 2007). The entered data also include variables related to the physical conditions in the river such as flow rate, temperature, cross-sectional area; and numerical parameters pertaining to the simulation (integration time, step size, method, etc.). The data was combined with the GIS system and transferred to Matlab™ platform for simulation. The simulations run on the Matlab platform determine the predictions of water quality along the river. Simulation results are relayed back to the GIS platform, and combined with the geographical data for display and analysis. The GUI was coded in Visual Basic™. The software allows the ArcMap and Access package programs to run interactively. This was accomplished through interlinking the ArcMap with Access (mdb) files, thus the data can be handled interactively. All graphics and tables were created from 'mdb' files.

In the previous work reported by Marsili–Libelli *et al.* (2002) data was transferred between the platforms by special 'avenue' script. This was appropriate in their case because the GIS platform that they used, ArcView (ESRI, 1996a), has a procedural language called Avenue (ESRI, 1996b) to define "scripts" that can implement the dynamic data exchange (DDE) procedure. However, the ArcMap™ 9.1 used here reads 'txt' files, so the data conveyed in ASCII format from Matlab are known. This data is then converted into dimensional variables in Visual Basic to be represented in tabulated form. For this procedure, the following SQL statements were used. These database connection statements make the data such that it can be viewed in graphs and tables, and also be used for color coding of the river information in GIS system.

```
Set m_pAdoCon = New ADODB.Connection
        m_pAdoCon.Open
"Provider=Microsoft.Jet.OLEDB.4.0; Data
            Source=C:\...\....mdb;Persist Security Info=False"
        Set pRecset = New ADODB.RecordSet
```

First of these statements opens database connections, second shows 'mdb' file path, name and table; and the third one starts the actual connection procedure. With these SQL statements, data become interconnected to ArcMap tables.

All windows and menus of the GUI, which are illustrated in the following figures, were designed in Visual Basic editor of ArcMap. The opening menu of the program is depicted in Fig. 2 together with the input sheet for entering initial water quality conditions. The table on the left hand side of the window lists the water quality variables that can be monitored on the screen. Prior to any run for simulation and prediction, the user is expected to enter the initial water quality conditions at starting point of working area where the simulation will begin. If there is a point source to the river, it can also be taken into account and respective values can be entered via additional input sheets that will open.

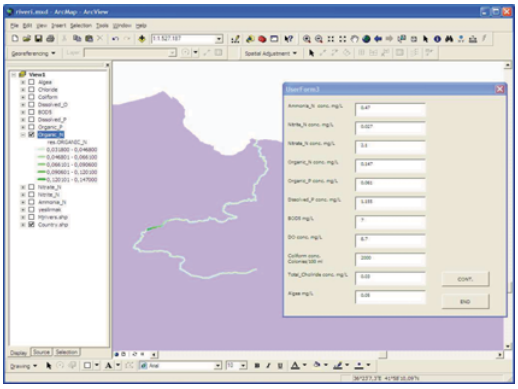


Fig. 2. View of the opening menu of the software (with the map of Yesilirmak river indicating the study area, and user input sheet).

Once the simulation is run, the user can select any variable from the list, shown in left hand side of the menu, to be displayed in table or graphical form.

The working area was divided into 100 parts of equal length to illustrate water quality variables, and thus the user can follow the concentration of the selected quality variable in different color at desired locations on map. The geographical point where the variables are sought is selected by the movement of the mouse along the river displayed on map. It then becomes possible to follow the water quality in terms of the selected pollutants along the river. For example, Fig. 3 depicts the change in the ammonia nitrogen concentration following a point source. With this feature, the simulation results are linked to the GIS database, and thus the user can easily follow the spatial distribution of the major constituents of river water quality. It is also possible to display more than one quality variable in graphical or table form at any location indicated.

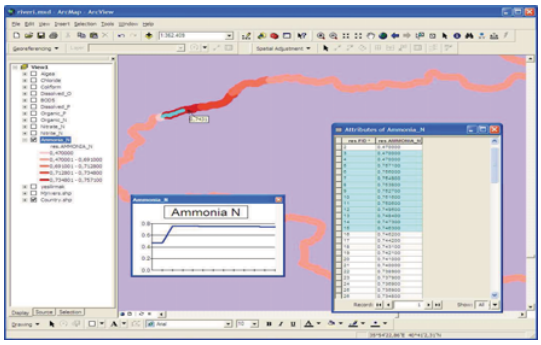


Fig. 3. Water quality display in table and graph form on the map.

In Fig. 3 the list on the left hand side shows the water quality variables considered. The user can select a location on the river map by moving the mouse, and if it is a point on the river the data table associated with this particular location opens on the screen. On the other hand, if the user scans a

region along the river, the software allows the user to see the changes in the concentration of the selected quality variables along this site by different colors. The color intensity on the map changes from light to dark with increased concentration, and this feature makes keeping track of water quality very easy. The user can select the concentration range (maximum and minimum values) and the number of intervals between. The color codes corresponding to those selected concentrations may be also determined by the user. If no choice has been made, the software picks up the maximum and minimum concentration values encountered in the simulations, and allocates five intermediate color codes (as default) to 5 intermediate values between the maximum and minimum. This feature of the GUI is illustrated in Fig. 4. It is also possible to see the changes in concentration in graphical form.

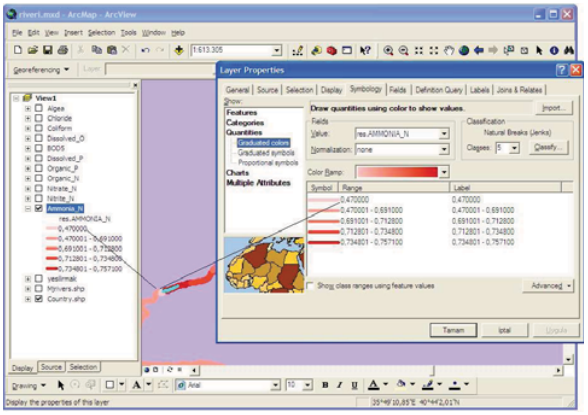


Fig. 4. Selection of color codes between maximum and minimum concentration intervals.

3. TESTING ON YESILIRMAK RIVER IN TURKEY

The software developed was tested with off line data collected from field studies around the city of Amasya along Yesilirmak river in Turkey. Yesilirmak is one of the major rivers in Turkey with 519 km length, and a basin of 36114 km^2 comprising 4.63 % of the territorial area of Turkey. Fig. 5 shows the study area on the river map. Pollution level in Yesilirmak affects the agricultural and rural development directly by distorting the ecological balance. The basin is a predominantly rural area and suffers from quite high level of pollution, in particular from agriculture, urban and industrial sources. Water quality in the river is classified in III and IV level according to the Water Pollution Control Act of Turkey, when physical, chemical, organic and bacteriological parameters are considered. An interactive river management decision support system for the region in order to protect the river from pollution becomes important for sustainable development in the future. Therefore, Yesilirmak was chosen as the study area, where our previous studies had also been concentrated.

The concentrations of ten water-quality constituents indicative of the level of pollution in the river were determined either on-site by portable analysis systems or in laboratory after careful conservation of the samples. For determination of dissolved oxygen, YSI Model 51/B portable oxygen meter in compliance with the Turkish Standard-TS 5677 were used. Nitrite, nitrate and ammonia forms of nitrogen were analyzed with HACH (Model DR2000) portable spectrophotometer. Total nitrogen was determined by Kjeldahl method. The organic nitrogen was calculated as the difference between the total nitrogen and the sum of ammonia, nitrite and nitrate forms. Phosphorous was analyzed by the methods of colorimetric ascorbic acid amino reduction and molibdo vanado phosphate (Greenberg, 1992) in the same spectrophotometer. BOD analysis and coliform analysis were done in the laboratory, after careful transportation of the samples, with manometric method in HACH spectrophotometer, and with multiple tubes and filtering method respectively. The chloride analysis was done in HACH spectrophotometer for free chloride and total chloride. Out of the 11 state variables, 10 were determined from field measurements. Only a representative data for the concentration of algae was taken from literature (Brown, 1987).

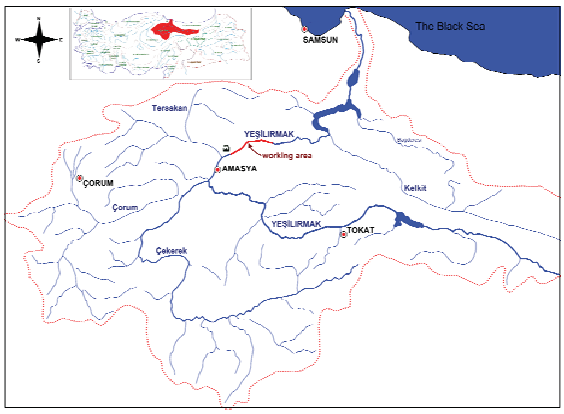


Fig. 5. Yesilirmak river basin and the study area.

During the dynamic sample collection period, the effluent from the wastewater treatment plant of a baker's yeast production plant was being discharged right beyond the starting point. Therefore, the results of the study indicated the extent of the pollution caused by the discharge from this industrial plant. In the simulations, addition of this discharge was considered as a continuous disturbance to the system, and its effect on the water quality, thus, was determined. Table 1 gives the characteristics of discharge from this local industrial plant.

Water quality data was collected for a 36.5 kms long section of the river adjacent to the city of Amasya. The study area started from the location 8.9 kms east of the city center where a baker's yeast plant was situated. The treated wastewater of this plant was considered as a point source to the river. The river water was sampled at 7 different locations in the downstream direction towards Durucasu gauging station of State Hydraulic Works (DSI) and the town of Tasova.

The initial conditions of the river and the characteristics of the point source as measured from points 1 and 2 indicated on Fig. 6 were introduced into the software, and dynamic simulation was run.

Table 1. Characteristics of discharge from the local baker's yeast plant

Variables	Waste water of baker's yeast plant
Temperature (^oC)	25.3
Flow (m^3/s)	0.25
Ammonia-N (mg/L)	27.4
Nitrite-N (mg/L)	1.3
Nitrate-N (mg/L)	52
Organic-N (mg/L)	0
Organic-P (mg/L)	0.52
Dissolved-P (mg/L)	12.4
BOD (mg/L)	210
DO (mg/L)	7.2
Coliform, (colonies/100 ml)	2900
Chloride (mg/L)	0

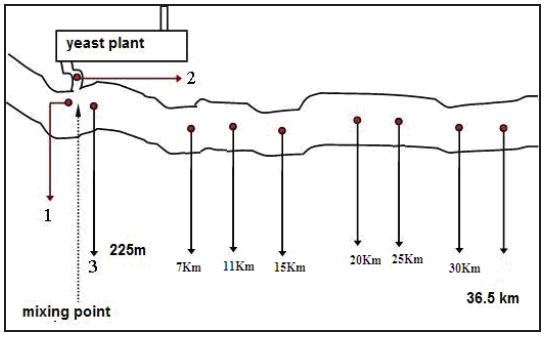


Fig. 6. Experimental study area for model verification, and sample collection points (distances indicated are measured from point 1).

Predictions from the software were compared to field data for a section of 36.5 kms of the river after the point source. Measured and predicted profiles of the pollution variables are shown in the following figures. Fig. 7, 8 and 9 reveal the pollution load due to the point source, and indicate that after some distance from the point where the effluent enters, remarkable recovery was observable. Fig. 10 shows the change in dissolved oxygen concentration along the study area. The points in the Figs. 7-8 indicate measurements whereas the continuous lines are predictions from the model.

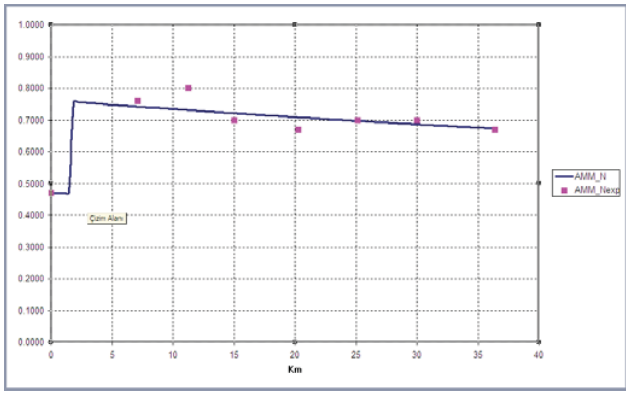


Fig. 7. Ammonia nitrogen profile.

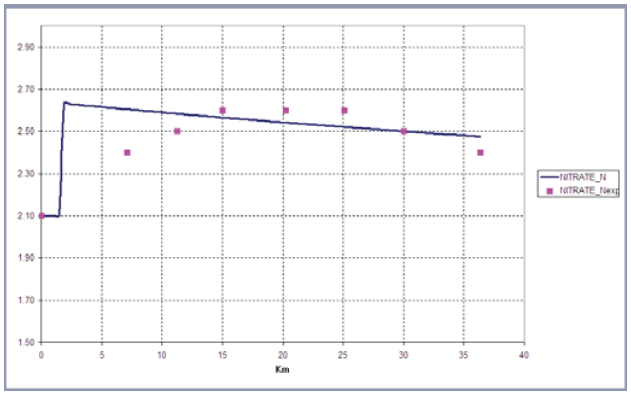


Fig. 8. Nitrate nitrogen profile.

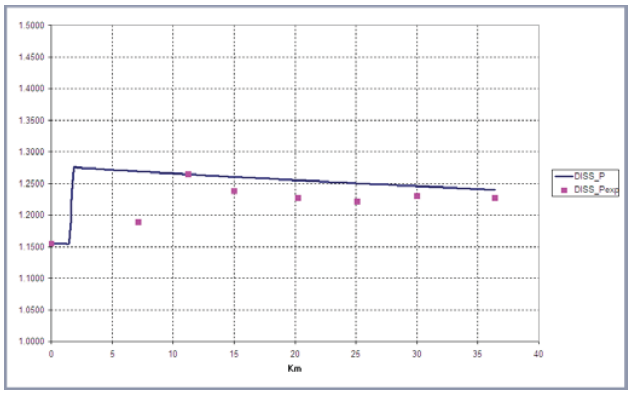


Fig. 9. Dissolved phosphorus profile.

For quantitative evaluation and comparison, Absolute Average Deviation (AAD) values were calculated. Table 2 indicates that, except nitrite nitrogen and chloride, the predicted values of all quality variables are in compliance with measured values.

Fig. 11 presents the predicted water quality results in tabulated form as a function of geographical space indicated on the first row. The columns represent the water quality variables predicted. The water quality data displayed can be viewed in graphical form or as color coded displays on the river map.

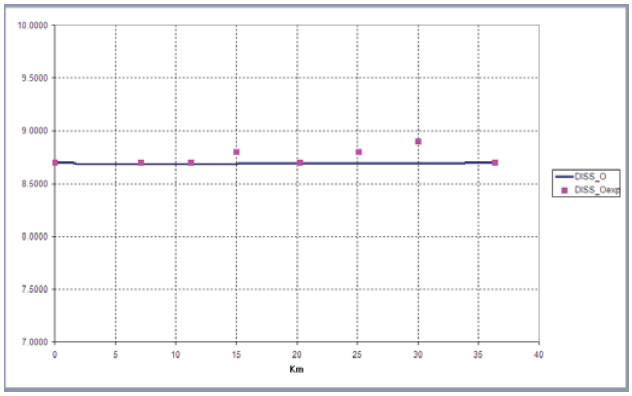


Fig. 10. Dissolved oxygen profile.

Table 2. Absolute Average Deviation (AAD) values for comparison of pollution variables

Water Quality Variables	(AAD %)
Ammonia Nitrogen	2.86
Nitrite Nitrogen	29.59
Nitrate Nitrogen	2.71
Organic Nitrogen	9.01
Organic Phosphorus	2.09
Dissolved Phosphorus	1.89
BOD	5.49
Dissolved Oxygen	0.64
Coliform	6.87
Chloride	20.19

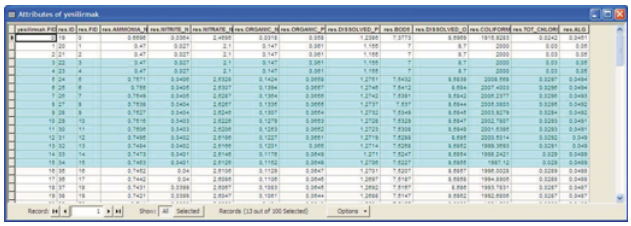


Fig. 11. Simulation results in tabulated form.

4. CONCLUSION

Although many models have been developed, they appear to be available to limited number of professionals who are capable of using and interpreting water quality simulation models. However, increased awareness in surface water pollution dictates that these models be used by non-experts who may be interested in knowing the consequences of various scenarios on river pollution. Availability and affordability of GIS systems offer alternative solutions to this problem.

Starting from this point, a software integrating a Geographical Information System and a water quality model in a single convenient package has been developed in this study. The effects of a discharge on the river can be predicted by simulation and the results are displayed on the map. The software was tested off-line with data collected from field measurements on Yesilirmak river in Turkey.

The integration strategy developed and the GUI created provide an interactive environment for the user and will help decision making process in river basin management systems, and can be fairly easily adapted to other rivers.

The results indicated that the model was able to satisfactorily estimate the water quality along the downstream section of a point load.

In our ongoing work, the software has been implemented for real time applications, and these results will be reported later.

REFERENCES

Berber, R., Yuceer, M. and Karadurmus, E. (2009). A parameter identifiability and estimation study in Yesilirmak River. *Water Science and Technology*, 59(3), 515–521.

Brown, L. C. (1987). Uncertainty analysis in water quality modelling using QUAL2E, EPA/Technical Report Data, Athens, GA.

Cianchi, P., Marsili–Libelli, S., Burchi, A. and Burchielli, S. (2000). Integrated river quality management using internet technologies. *Watermatex*. Gent(B), 18-20.

Cox, B.A. (2003). A review of available in-stream water quality models and their applicability for simulating dissolved oxygen in lowland rivers. *The Science of the Total Environment.* 314-316, 335-377.

Davies, H. and Neal, C. (2004). GIS-based methodologies for assessing nitrate, nitrite and ammonium distribution across a major UK basin, the Humber. *Hydrology and Earth System Sciences.* 8(4), 823-833.

ESRI (1996a) ArcView GIS. *The geographic information system for everyone*. Environmental Systems Research Institute, USA.

ESRI (1996b) ArcView Spatial Analyst. *Advanced spatial analysis using raster and vector data*. Environmental Systems Research Institute.

Ganapathy, C. and Ernest, A.N.S. (2004). Water quality assessment using web based GIS and distributed database management systems. *Environmental Informatics Archives*. 2, 938-945.

Greenberg, A.E., Clesceri, L.S., Lenore, S. and Eaton, A.D. (1992). Standard methods for the examination of water and wastewater. *American Public Health Assoc.* Washington D.C.

Karadurmus, E. and Berber, R. (2004). Dynamic simulation and parameter estimation in river streams. *Environmental Technology*. 25, 471-479.

Marsili–Libelli, S., Caporali, E., Arrighi, S. and Becattelli, C. (2001). A georeferenced water quality model. *Water Sci. Tech.* 43(7), 223-230.

Marsili–Libelli, S., Pacini, G., Barresi, C., Petti, E. and Sinacori, F. (2002). An interactive georeferenced water quality model. *Hydroinformatics: Proc. of the 5th Int. Conf. on Hydroinformatics*, Cardiff, UK.

Mitchell, A. (1999). *The ESRI Guide to GIS Analysis*. Vol. 1, Esri Press, USA.

Rauch, W., Henze, M., Koncsos, L. Reichert, P., Shanahan, P., Somlyody, L. and Vanrolleghem, P. (1998). River water quality modelling: I. State of the art. *IAWQ Biennial Int. Conf.* Vancouver-Canada, 21-26.

Ruelland, D.,Billen, G., Brunstein, D., Garnier, J. (2007). SENEQUE: A multi-scaling GIS interface to the Riverstrahler model of the biogeochemical functioning of river systems. *Science of the Total Environment.* 375, 257-273.

Shanahan, P., Henze, M., Koncsos, L. Rauch, W., Reichert, P. Somlyody, L. and Vanrolleghem, P. (1998). River water quality modelling: II. Problems of the art. *IAWQ Biennial Int. Conf.* Vancouver-Canada, 21-26.

The MathWorks Inc. Natick- MA. (2003).

Yuceer, M., Karadurmus, E. and Berber, R. (2007). Simulation of River Streams: Comparison of a New Technique to QUAL2E. *Mathematical and Computer Modelling*. 46, 292–305.

Unscented Kalman Filter state and parameter estimation in a photobioreactor for microalgae production ⋆

Giancarlo Marafioti [*] **Sihem Tebbani** [**] **Dominique Beauvois** [**]
Giuliana Becerra [**, ***] **Arsene Isambert** [***] **Morten Hovd** [*]

[*] *Department of Engineering Cybernetics, Norwegian University of Science and Technology, N-7491 Trondheim, Norway*
{giancarlo.marafioti, morten.hovd}@itk.ntnu.no
[**] *Control Department, SUPELEC, Plateau de Moulon, 91192 Gif sur Yvette, France* {sihem.tebbani, dominique.beauvois, giuliana.becerra}@supelec.fr
[***] *LGPM, Ecole Centrale Paris, 92295 Chatenay-Malabry, France*
arsene.isambert@ecp.fr

Abstract: Microalgae have many applications such as the production of high value compounds (source of long-chain polyunsaturated fatty acids, vitamins, and pigments), in energy production (e.g. photobiological hydrogen, biofuel, methane) or in environmental remediation (especially carbon dioxide fixation and greenhouse gas emissions reduction). However, the photobioreactor microalgae process needs complex and costly hardware sensors, especially for biomass measurement. Thus, state and parameter estimation seems to be a critical issue and is studied in this paper in the case of a culture of the microalga Porphyridium purpureum. This paper is an extension of the previous work of Becerra-Celis et al. (2008) where the principal objective is to design a biomass estimator of this microalga production in a photobioreactor based on the total inorganic carbon measurement.
Unscented Kalman filtering is applied to estimation of states and model parameters, producing better performances in comparison with Extented Kalman filtering. Numerical simulations in batch mode, and real-life experiments in continuous mode have been carried out. Corresponding results are given in order to highlight the performance of the proposed estimator.

Keywords: Unscented Kalman Filter, state and parameter estimation, microalgae photobioreactor

1. INTRODUCTION

In chemical and biochemical processes, often chemical reactions have to be monitored and controlled using different sensor measurements. Typically, measurements of reactant and product concentrations, operating temperatures, pressures, and other parameters are needed. In general, a measurement has to be reliable, i.e. it has to be available and accurate. However, there are several reasons why required measurements may not be reliable. Some of such reasons are the impracticability of building an appropriate sensor due to lack of technology, the difficulty to position the sensor, the associated cost. In such cases, an attempt to use estimation techniques may be done.

Dochain (2003) presents an interesting overview of available results on state and parameter estimation in chemical and biochemical processes. A comparison of several traditional state and parameter estimation approaches is given, discussing pros and cons in different cases, and describing how the most common implementation problems are solved (see Dochain (2003) and references therein). In this work particular attention is given to Kalman filtering. Its application to nonlinear systems is tipically implemented by the well known and widely used Extended Kalman Filter (EKF). Even though there are issues due to the inherent linearization procedure in the algorithm, the scientific and industrial communities have obtained successful EKF applications. However, it is the authors' opinion that in systems with strong nonlinearities it could be interesting to exploit the benefits of the Unscented Kalman Filter (UKF). Thus, the UKF is introduced as a valid EKF alternative, and it is shown how to obtain improvement due to its implementation flexibility, extending then its applicability. The first works introducing the Unscented transformation idea and the UKF algorithm are Julier and Uhlmann (1996), and Julier and Uhlmann (1997). In Wan and Van Der Merwe (2000) several UKF algorithms are described, which could be used for state estimation, parameter estimation, and joint state and parameter estimation. This work aims to present UKF advantages in terms of performance and implementation ease, compared to the EKF. The work of Becerra-Celis et al. (2008) is considered as starting point, where it is shown how to implement an

⋆ The authors wish to acknowledge support from the French-Norwegian research cooperation project AURORA

EKF for state estimation in a photobioreactor. Moreover, experimental data is used to validate the results.

Next section explains how to use the UKF for joint state and parameter estimation. Section 3 describes the photobioreactor used for microalgae production, and its model. Section 4 analyzes the application of the UKF to the photobioreactor. In addition, it is shown that when parameter estimation is considered, the UKF produces improved results, particularly when experimental data, collected from continuous cultures, is used. Finally, conclusions and future work are presented.

2. STATE AND PARAMETER ESTIMATION

Consider the following nonlinear system

$$\begin{aligned}\dot{\xi}(t) &= g(\xi(t), u(t), v_\xi(t)) \\ y(t) &= l(\xi(t)) + n(t)\end{aligned} \tag{1}$$

where ξ is the state vector, u is the input vector, y is the measurement vector, v_ξ is the process noise vector, n is the measurement noise vector, of appropriate dimensions, respectively. Simply stated, the state estimation problem consists to reconstruct the state vector knowing the measurement and input vectors, some information on noise distributions, and the nonlinear functions $g(\cdot)$ and $l(\cdot)$. When the parameters used in the model (1) are uncertain the estimation problem may be more difficult to solve. One method to obtain sufficiently good estimates is to make the estimator algorithm robust with respect to the parameter variations. Another method is to try to estimate the uncertain parameters improving the model accuracy. However, joint parameter and state estimation may lead to observability problems. A general framework to introduce the parameter estimation is to extend the state with the uncertain parameters vector and then estimate the augmented state. In cases where the actual parameters are slowly varying, it is common to model the parameters vector η as a random walk driven by a white noise process $v_\eta(t)$. Thus the following differential equation

$$\dot{\eta}(t) = v_\eta(t) \tag{2}$$

is used to augment the system (1). Associating then a relatively small covariance to $v_\eta(t)$, it is possible to consider the slowly varying nature of the parameters. The section below describes the particular UKF used in this work. The algorithm describes the case of joint parameter and state estimate. The equations are still valid if only state estimate is considered. However, the procedure to augment the state vector and covariance matrix, with the parameter vector and its covariance matrix, respectively, must be omitted.

2.1 Unscented Kalman Filter algorithm

Consider the following discretization of the system (1-2)

$$\begin{aligned}\xi_{k+1} &= f(\xi_k, u_k, v_k^\xi, \eta_k) \\ \eta_{k+1} &= \eta_k + v_k^\eta \\ y_k &= h(\xi_k, \eta_k) + n_k\end{aligned} \tag{3}$$

obtained using an appropriate numerical integration routine. To estimate jointly the state and parameters of the system (3), the following augmented vector is defined:

$$\hat{x}_k^a = [\hat{x}_k',\ v_k{}']' \tag{4}$$

where $\hat{x}_k = [\hat{\xi}_k',\ \hat{\eta}_k']'$ has as elements the state and parameter estimates, respectively. The vector $v_k = [v_k^{\xi'}, v_k^{\eta'}]'$ contains the process noises in the evolution of ξ and η. Analogously, the augmented covariance matrix is defined as

$$P_k^a = \begin{bmatrix} P_k^x & P_k^{x,v} \\ P_k^{v,x} & P_k^v \end{bmatrix} \tag{5}$$

where P_k^x consists of the state and parameter error covariances, while P_k^v includes the process noise covariance associated to state and parameters. In addition, the off diagonal entries are cross covariance terms represented by the notation $P_k^{\cdot,\cdot}$. Obviously, all elements of $\hat{x}_k^a$ and P_k^a are of appropriate dimensions.

Given the initial conditions

$$\hat{x}_0^a = \begin{bmatrix} \hat{x}_0 \\ 0_v \end{bmatrix}, \qquad P_0^a = \begin{bmatrix} P_0^x & 0 \\ 0 & P_0^v \end{bmatrix}, \tag{6}$$

the dimension L of the augmented system state $\hat{x}^a$, and the following scalar weigths W_i

$$W_0^{(m)} = \lambda/(L+\lambda) \tag{7}$$

$$W_0^{(c)} = \lambda/(L+\lambda) + (1-\alpha^2+\beta) \tag{8}$$

$$\begin{aligned}W_i^{(m)} &= W_i^{(c)} \\ &= 1/[2(L+\lambda)]\end{aligned} \tag{9}$$

for $i = 1, \ldots, 2L$, $\lambda = \alpha^2(L+\kappa) - L$, and where α, β, κ are parameters to be chosen.

For $k = 1, \ldots, \infty$

- Calculate the sigma points defined as

$$\begin{aligned}\left(\mathcal{X}_{k-1}^a\right)_0 &= \hat{x}_{k-1}^a \\ \left(\mathcal{X}_{k-1}^a\right)_j &= \hat{x}_{k-1}^a + \gamma \left(\sqrt{\mathrm{P}_{k-1}^a}\right)_j \\ \left(\mathcal{X}_{k-1}^a\right)_{j+L} &= \hat{x}_{k-1}^a - \gamma \left(\sqrt{\mathrm{P}_{k-1}^a}\right)_j\end{aligned} \tag{10}$$

  where $\left(\sqrt{\mathrm{P}_{k-1}^a}\right)_j$ is the j-th column ($j = 1, \ldots, L$) of the square root of the augmented covariance matrix (5) at the previous time step. The parameter $\gamma = \sqrt{L+\lambda}$ can be interpreted as a scaling factor used to move the position of sigma points around the mean value $\hat{x}_{k-1}^a$. Finally, the sigma points are regrouped in the following matrix of L rows and $2L+1$ columns:

$$\mathcal{X}_{k-1}^a = \begin{bmatrix} \mathcal{X}_{k-1}^x \\ \mathcal{X}_{k-1}^v \end{bmatrix} \tag{11}$$

  where $\mathcal{X}_{k-1}^x$ contains the sigma point rows associated to state and parameters, and $\mathcal{X}_{k-1}^v$ the sigma point rows associated to the state and parameters process noises.

- Propagate the sigma points through the nonlinear dynamics F[·], and compute the predicted state estimate, where the index i is used to select the appropriate sigma point column:

$$\mathcal{X}_{k|k-1} = \mathrm{F}\left[\mathcal{X}^a_{k-1}, u_{k-1}\right] \tag{12}$$

$$\hat{x}^-_k = \sum_{i=0}^{2L} W^{(m)}_i \mathcal{X}_{i,k|k-1} \tag{13}$$

- Compute the predicted covariance:

$$P^-_k = \sum_{i=0}^{2L} W^{(c)}_i \left[\mathcal{X}_{i,k|k-1} - \hat{x}^-_k\right]\left[\mathcal{X}_{i,k|k-1} - \hat{x}^-_k\right]' \tag{14}$$

- Using the predicted mean (13) and covariance (14), recompute a new set of sigma points as defined in (10-11):

$$\mathcal{X}^-_k = \left[\mathcal{X}^{x\prime}_k, \mathcal{X}^{v\prime}_k\right]' \tag{15}$$

- Instantiate the new sigma points through the observation model H[·], and calculate the predicted measurement:

$$\mathcal{Y}_{k|k-1} = \mathrm{H}\left[\mathcal{X}^-_k\right] \tag{16}$$

$$\hat{y}^-_k = \sum_{i=0}^{2L} W^{(m)}_i \mathcal{Y}_{i,k|k-1} \tag{17}$$

- Obtain the innovation covariance and the cross covariance matrices:

$$P_{\tilde{y}_k\tilde{y}_k} = \sum_{i=0}^{2L} W^{(c)}_i \left[\mathcal{Y}_{i,k|k-1} - \hat{y}^-_k\right]\left[\mathcal{Y}_{i,k|k-1} - \hat{y}^-_k\right]' + P^n \tag{18}$$

$$P_{y_k x_k} = \sum_{i=0}^{2L} W^{(c)}_i \left[\mathcal{X}_{i,k|k-1} - \hat{x}^-_k\right]\left[\mathcal{Y}_{i,k|k-1} - \hat{y}^-_k\right]' \tag{19}$$

where P^n is the measurement noise covariance;

- Perform the measurement update using the regular Kalman filter equations:

$$\mathcal{K}_k = P_{y_k x_k} P^{-1}_{\tilde{y}_k\tilde{y}_k} \tag{20}$$

$$\hat{x}_k = \hat{x}^-_k + \mathcal{K}_k\left(y_k - \hat{y}^-_k\right) \tag{21}$$

$$P^x_k = P^-_k - \mathcal{K}_k P_{\tilde{y}_k\tilde{y}_k}\mathcal{K}'_k \tag{22}$$

In (12), F[·] is the modified nonlinear dynamics of (3). The changes are made to consider the discretization and the augmented state, and also to guarantee the proper propagation of each sigma point. Analogously, in (16) H[·] is the modified observation function. Due to the fact that measurement noise is assumed additive with zero mean, it is possible to write (18). Thus, the algorithm computational complexity is reduced because there is no need to associate more sigma points. Regarding filter design parameters, in most cases typical values are $\beta = 2$, and $\kappa = 0$ or $\kappa = 3 - L$, leaving only the parameter α as free parameter. Moreover, considering that $1 \leq \alpha \leq 10^{-4}$ the tuning of the UKF becomes simpler. For a finer tuning and a more accurate description about the meaning of the UKF parameters one can refer to Wan and Van Der Merwe (2000). Finally, in (11) and (15) a square root of a matrix has to be calculated, thus an appropriate algorithm must be used, for instance the Cholesky factorization.

3. PHOTOBIOREACTOR FOR MICROALGAE PRODUCTION

3.1 Strain and growth conditions

The photobioreactor is used to produce the red microalgae *Porphyridium purpureum* SAG 1830-1A obtained from the Sammlung von Algenkulture Pflanzenphysiologister Institut Universität Göttingen, Germany. The strain is growth and maintained on Hemerick medium (Hemerick (1973)). The pH of the Hemerick medium is adjusted to 7.0 before autoclaving it for 20 minutes at 121 °C. Cultures are maintained at 25 °C in 500 ml flask containing 400 ml culture under continuous light intensity of 70 $\mu Em^{-2}s^{-1}$ and aerated with air containing 1% (v/v) CO_2 at 100 rpm on an orbital shaker. During the exponential growth phase, within an interval of two weeks, 200 ml of culture are transferred to a new flask containing fresh medium.

3.2 Culture conditions and measurements

Figure 1 illustrates the photobioreactor diagram where the growth of cultures is performed. The bubble column photobioreactor has a working height of 0.4 m and a diameter of 0.1 m. The total culture volume is 2.5 l, and the cylindrical reactor, made of glass, has an illuminated area of 0.1096 m^2. To agitate the culture an air mixture with 2% (v/v) CO_2 is continuously supplied at a flow rate of 2.5 *V.V.H* (gas volume per liquid culture volume per hour). 0.22 μm Millipore filters, appropriate valves and flowmeters are used to filter and to control the air flow rate entering the photobioreactor. Four OSRAM white fluorescent tubes (L30W/72) and three OSRAM pink fluorescent tubes (L30W/77) are arranged around the bubble column as an external light source. The incident light intensity on the reactor surface is measured at ten different locations with flat surface quantum sensors (LI-COR LI-190SA). The average light intensity is computed by the weighted average of all measurements. The optimal value of irradiance on surface for the reactor is 120 $\mu Em^{-2}s^{-1}$. A transparent jacket connected to a thermostat unit allows the temperature control, which is regulated at 25 °C. Other sensors are a pH sensor (Radiometer Analytical) and a dissolved oxygen sensor (Ingold type 170). A sampling port is applied to the top of the column, from where samples for off line analysis are collected after 6, 8, and 12 hours. The number of cells is counted using an optical microscope ZEISS Axioplan-2 on Malassez cells. The total inorganic carbon (T.I.C.) in the culture medium is calculated by gas phase chromatography. This method, proposed by Marty et al. (1995), is used to measure low inorganic carbon concentrations down to ($10^{-6} mol\ l^{-1}$) within an accuracy of 10%.

3.3 Mathematical model

In this work the bioprocess model presented in Baquerisse et al. (1999) is used. It consists of two sub models, one describing the growth kinetics, and one representing the gas-liquid mass transfer in the photobioreactor. This

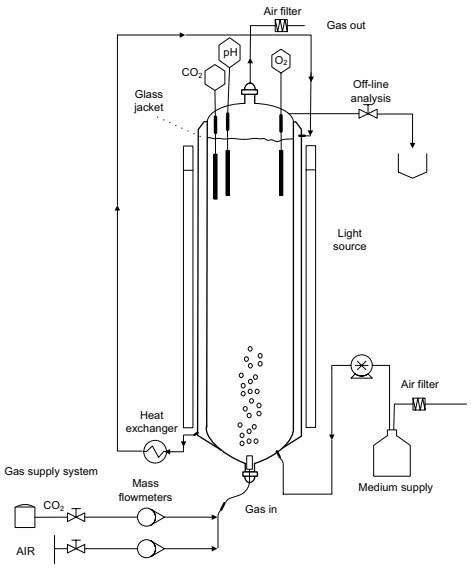


Fig. 1. Photobioreactor diagram.

results in two differential equations describing the state of the reactor:

$$\begin{aligned}\frac{dX}{dt} &= \frac{F_{in}}{V}X_{in} + \mu X - \frac{F_{out}}{V}X \\ \frac{d[TIC]}{dt} &= \frac{F_{in}}{V}[TIC]_{in} - \frac{F_{out}}{V}[TIC]_{out} - \mu\frac{X}{Y_{X/S}} \\ &\quad - mX + k_L a\left([CO_2^*] - [CO_2]\right) \end{aligned} \tag{23}$$

where X is the biomass, and $[TIC]$ is the inorganic carbon concentration associated with cell density increase. The subscripts $[\cdot]_{in}$ and $[\cdot]_{out}$ indicate quantities flowing into, and out from the reactor, respectively. V is the culture volume, and F is the medium flow rate. The mass conversion yield is defined by $Y_{X/S}$, m is the maintenance coefficient, and $k_L a$ is the gas-liquid transfer coefficient. The carbon dioxide concentration in the medium fresh is defined as:

$$[CO_2^*] = \frac{PCO_2}{\mathcal{H}} \tag{24}$$

where PCO_2 is the partial pressure of carbon dioxide, and $\mathcal{H}$ is the Henry's constant for Hemerick medium. Moreover, the carbon dioxide concentration in the medium is given by:

$$[CO_2] = \frac{[TIC]}{\left[1 + \frac{K_1}{[H^+]} + \frac{K_1 K_2}{[H^+]^2}\right]} \tag{25}$$

where K_1, K_2 are kinetics constants, and $[H^+]$ is defined as:

$$[H^+] = 10^{-pH} \tag{26}$$

representing the hydrogen ions concentration in the culture media.

In addition, a light transfer model is considered, which describes the evolution of incident and outgoing light intensity:

$$E = \frac{(I_{in} - I_{out})A_r}{VX} \tag{27}$$

$$I_{out} = C_1 I_{in} X^{C_2} \tag{28}$$

where E is the light "energy" accessible per cell, I_{out} is the outgoing light intensity, I_{in} is the ingoing light intensity. C_1, C_2 are constants depending on the reactor geometry, and A_r is its area.

The light intensity and the total carbon concentration influence the specific growth rate, defined as

$$\mu = \mu_{max}\frac{E}{E_{opt}}e^{\left(1-\frac{E}{E_{opt}}\right)}\frac{[TIC]}{[TIC]_{opt}}e^{\left(1-\frac{[TIC]}{[TIC]_{opt}}\right)} \tag{29}$$

where μ_{max}, E_{opt}, and $[TIC]_{opt}$ are model parameters identified from the batch data experiments. Finally, in (29) substrates limitation effect is taken into account.

3.4 Batch and continuous operating conditions

The photobioreactor can work in two different operating conditions, batch mode and continuous mode. In batch mode:

$$F_{in} = F_{out} = 0; \qquad [TIC]_{in} = 0; \qquad X_{in} = 0. \tag{30}$$

In continuous mode, instead:

$$F_{in} = F_{out} \neq 0. \tag{31}$$

3.5 Model parameters

The model parameters used in this work are the ones identified in Becerra-Celis et al. (2008). For more details on the system identification procedure the reader is referred to their work. Tables 1 and 2 contain the parameters for the microalgae and the total inorganic carbon dynamics, respectively.

Table 1. Model parameters for *Porphyridium purpureum* at 25 °C.

Parameter	Unit	Value
μ_{max}	h^{-1}	0.0337
E_{opt}	$\mu Es^{-1}(10^9 cell)^{-1}$	1.20
$[TIC]_{opt}$	$mmolel^{-1}$	12.93
$k_L a$	h^{-1}	41.40
C_1		0.28
C_2		−0.55

Table 2. Model parameters for [TIC] dynamics.

Parameter	Unit	Value
K_1		$1.02 \cdot 10^{-6}$
K_2		$8.32 \cdot 10^{-10}$
m	$h^{-1} mmole(10^9 cell)^{-1}$	0.004
$Y_{X/S}$	10^9 *cell per mole TIC*	198.1
$\mathcal{H}$	*atm l mole*$^{-1}$	34.03

4. BIOMASS ESTIMATION

Controlling a photobioreactor has some difficulties associated to the implicit nonlinear and time varying nature of the system. There are also problems with the practicability

to find reliable online sensors able to measure the state variables (Shimizu (1996)). To overcome the lack of online sensors, Becerra-Celis et al. (2008) show how to implement an Extended Kalman Filter (EKF) to estimate the biomass for the photobioreactor, described in Section 3, using the measurement of T.I.C.. In this work, it is shown how the use of an UKF gives better performance, particularly for continuous cultures. There are several reasons for which the UKF may be considered as an EKF alternative. There is no linearization procedure in the UKF, as can be seen in section 2.1. This is relevant when strong nonlinearities are present in the process because no linearization error is introduced. It is straightforward to extend the state estimation to joint estimation, just augmenting the estimated vector and covariance matrix, while calculation of system derivatives with respect to the parameters, are required in an EKF algorithm.

4.1 UKF applied to the photobioreactor

Using the UKF, described in Section 2.1, the main objective is to estimate the biomass X in the photobioreactor of Section 3. Focusing on the two different working conditions defined in Section 3.4, it is observed how in batch mode the UKF has an excellent performance, which also is the case for the EKF designed in Becerra-Celis et al. (2008). This is due to the fact that the model parameters are identified in batch mode, and the measurements have a constant sampling time. A more complex scenario appears for continuous cultures. The model parameters are still the ones from the batch experiments, and the experimental data are collected at variable instant intervals. Due to the variable time steps, Becerra-Celis et al. (2008) implements a continuous discrete version of the EKF. In this work, this problem is tackled in two steps. Firstly, a zero order hold is applied to the measurements, secondly the standard UKF algorithm is properly modified. More in detail, the discrete UKF algorithm with an augmented state to consider parameter estimation and process noise is implemented. The sigma points are recomputed in (15) and then used to obtain the predicted measurement in (16-17). Those modifications give the possibility to use the discrete algorithm with the irregular measurement sampling time of the continuous culture case. Furthermore, the parameter μ_{max} in (29) is chosen to be estimated. Finally, despite the fact that a zero order hold is used to permit a discrete UKF implementation, the UKF accuracy and speed of convergence are improved with respect to the EKF ones.

4.2 Simulation results with experimental data

The following results show the efficiency of the proposed method when the photobioreactor works either in batch mode or in continuous mode. Figure 2 illustrates the convergence of the UKF in simulated batch mode, for which conditions (30) hold. In this case the nonlinear model (23) is discretized at sampling time $T_s = 0.5\ h$ and used to simulate the state of the process, starting from initial conditions $X_0 = 2.44 \cdot 10^9 cell/l$, $[TIC]_0 = 2.55\ 10^{-3} mole/l$. After that, $[TIC]$ is corrupted by additive Gaussian white noise with standard deviation $\sigma = 0.2\ 10^{-3} mole/l$, and used as measurement for the UKF. Thus, the state is

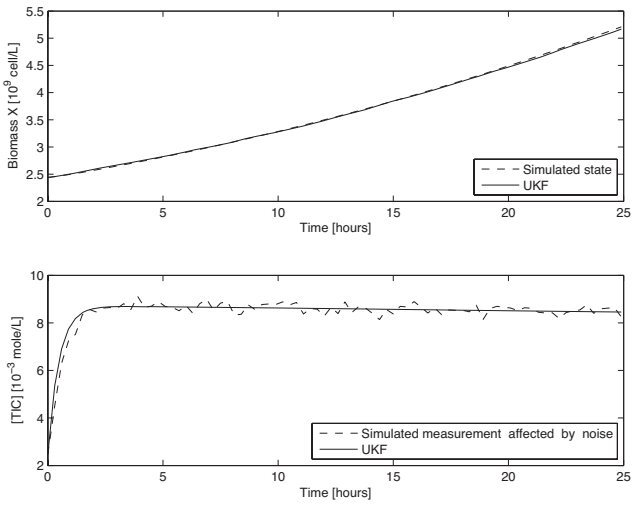
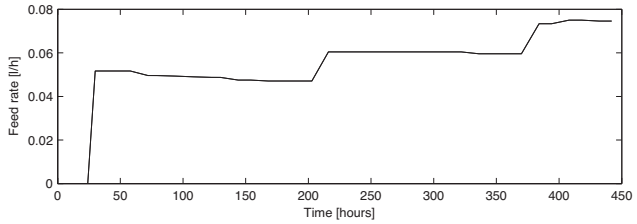
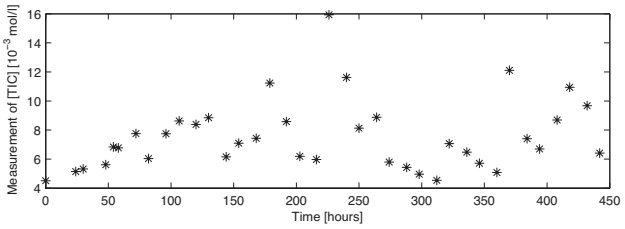


Fig. 2. UKF estimation for simulated batch mode.



(a) Culture medium feeding profile



(b) Measurement of [TIC]

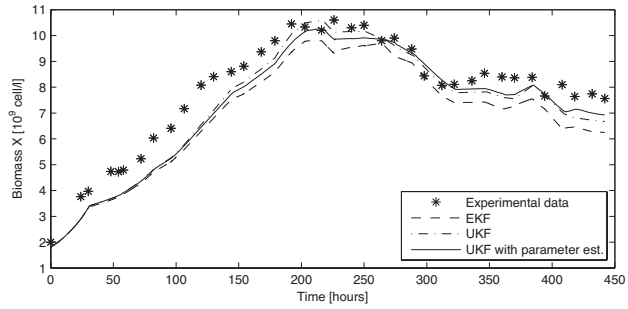
Fig. 3. Experimental data: input and output of the photobioreactor collected in continuous mode.

estimated successfully with excellent noise rejection in T.I.C.. The results obtained for continuous cultures are even more interesting. Initial conditions are $X_0 = 1.8 \cdot 10^9 cell/l$, $[TIC]_0 = 4.51 \cdot 10^{-3} mole/l$, and in addition real experiment data, presented in Figure 3, are used as input to the filters. The EKF designed in Becerra-Celis et al. (2008), the UKF with only state estimation, and the UKF with joint state and parameter estimation are simulated. The results obtained are shown in Figure 4 and here discussed. In Figure 4(a) it is noticeable how both UKF implementations have faster speeds of convergence than the EKF. In Figure 4(b) the state estimation error of the three different approaches are compared, and it is evident how the UKFs give smaller estimation errors. Moreover the mean squared error (MSE) between the biomass and its estimate is computed. Table 3 shows the MSE index, which is obtained averaging the MSE along the entire simulation period. From both figures it is noticeable how the UKF performs better than the EKF, and how the introduction

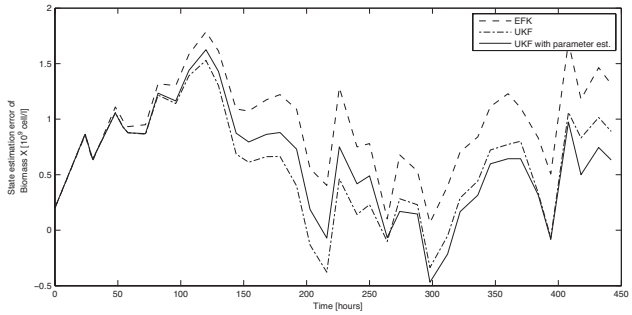
Table 3. Mean Squared Error Index

EKF	UKF	UKF with par. est.
13.60	6.12	6.12

of parameter estimation in the UKF improves the accuracy of the estimation in the final part (after 300 hours), although slightly reduces the speed of convergence. Since the parameters are identified from batch experiments, as shown in Becerra-Celis et al. (2008), adding parameter estimation may be useful when the photobioreactor is run in continuous mode. Figure 5 shows the evolution of the μ_{max} estimation, the estimated value is compared with the identified value and moreover the UKF error covariance is shown.



(a) Biomass estimate

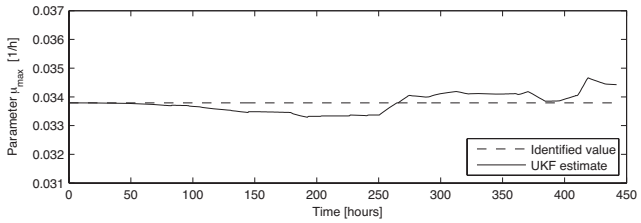


(b) State Estimation Error

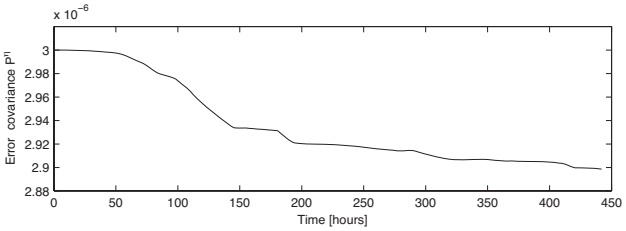
Fig. 4. Biomass estimation comparison for continuous cultures.

5. CONCLUSION

This work can be considered as an extension of Becerra-Celis et al. (2008), where the main objective is to design an efficient, reliable and applicable biomass estimator for a microalgae photobioreactor. A more recent nonlinear estimator (UKF) is used, obtaining improved results. In both batch and continuous mode, the approach presented produces a faster estimate convergence and a better estimate accuracy. The capacity and ease to introduce parameter estimation jointly with state estimation, the absence of linearization, the comparable computational complexity make the UKF an attractive estimator for nonlinear systems. The particular UKF framework described in section 2.1 showed to be well suited for the case when measurements, arriving at variable time instants, are subject to a zero holder filter. The extra set of sigma points calculated in (15) are needed to obtain a smoother estimate. Future work is needed to design a feedback based controller using



(a) UKF estimate and identified value of μ_{max}



(b) UKF error covariance of μ_{max}

Fig. 5. UKF parameter estimate and its covariance for continuous culture.

the UKF estimate, and to better understand which parameter would improve the filter performance without introducing observability issues. It is also interesting to explore how the UKF performs with other processes compared to the traditional methods in use in practice.

REFERENCES

Baquerisse, D., Nouals, S., Isambert, A., dos Santos, P.F., and Durand, G. (1999). Modelling of a continuous pilot photobioreactor for microalgae production. *Journal of Biotechnology*, 70(1-3), 335 – 342.

Becerra-Celis, G., Tebbani, S., Joannis-Cassan, C., Isambert, A., and Boucher, P. (2008). Estimation of microalgal photobioreactor production based on total inorganic carbon in the medium. *17th IFAC W.C., Seoul*.

Dochain, D. (2003). State and parameter estimation in chemical and biochemical processes: a tutorial. *Journal of process control*, 13, 801 – 818.

Hemerick, J. (1973). Culture methods and growth measurements. In J. Stein (ed.), *Handbook of Physiological Methods*, 250–260. Cambridge University Press, UK.

Julier, S. and Uhlmann, J.K. (1996). A general method for approximating nonlinear transformations of probability distributions. Technical report, .

Julier, S.J. and Uhlmann, J.K. (1997). A new extension of the kalman filter to nonlinear systems. In *Int. Symp. Aerospace/Defense Sensing, Simimulation and Controls*, 182–193.

Marty, A., Cornet, J., Djelveh, G., Larroche, C., and Gros, G. (1995). A gas phase chromatography method for determination of low dissolved CO_2 concentration and/or CO_2 solubility in microbial culture media. *Biotechnology Technique*, 9(11), 787–792.

Shimizu, K. (1996). A tutorial review on bioprocess systems engineering. *Computers and Chemical Engineering*, 20, 915–941.

Wan, E.A. and Van Der Merwe, R. (2000). The unscented kalman filter for nonlinear estimation. *Adaptive Systems for Signal Processing, Communications, and Control Symposium*, 153–158.

Dynamic model of NOx emission for a fluidized bed sludge combustor

S. Li[1], C. Cadet[1], P.X. Thivel[2], F. Delpech[2]

[1]GIPSA-lab, Dep Automatique, UMR 5216 CNRS-INPG-UJF
BP46, 38402 Saint Martin d'Heres Cedex, France (tel. +33(0)476826412,
e-mail: shi.li@lagep.univ-lyon1.fr, catherine.cadet@gipsa-lab.inpg.fr
[2]LEPMI, UMR 5631 CNRS-INPG-UJF
BP 76, 38402 38402 Saint Martin d'Heres Cedex (tel. +33(0)476826733,
e-mail: pierre-xavier.thivel@ujf-grenoble.fr, francoise.delpech@ujf-grenoble.fr

Abstract: Sludge incineration in fluidized bed is a very complex process, producing gaseous pollutants (carbon monoxide (CO) and nitrogen oxides (NOx)). The legislative norms need to be respected in spite of variations of sludge in composition and in quantity. The NOx formation, due to lots of chemical reactions from sludge nitrogen, is partly unknown. This paper deals with the design of a dynamic model of NOx emissions for a fluidized bed sludge combustor to be used in control strategy. The model is validated with industrial data, and the simulation results validate the simplification hypotheses, but need the reconstruction of sludge composition.

Keywords: sludge incineration, fluidized bed, NOx, chemical reaction, modelling, validation.

1. INTRODUCTION

The treatment and disposal of sewage sludge is an expensive and environmentally sensitive problem. It is also a growing worldwide problem since sludge production will continue to increase and since environmental quality standards become more stringent. Compared with landfill and agricultural compost, incineration presents some advantages: large volume reduction, stabilized ash production (heavy metals included) and toxic organic matters destruction. However to be economically viable, sludge has to be burned without fuel supply locally in wastewater treatment plant (Reimann, 1999).

Fluidized bed combustors are industrially largely used for coal, wastes and sewage sludge combustion (Werther and Ogada, 1999). The incineration process produces gaseous pollutants, mainly carbon monoxide (CO) and nitrogen oxides (NOx), which have to respect the legislative norms. Carbon monoxide formation can be limited by oxygen supply regulation, which is widely used in industrial plants. However, nitrogen oxides emissions should be controlled during combustion because its post-treatments are rather expensive. Linear dynamic models of coal combustion to be used in control strategy (Muir et al., 1997), (Bittanti et al., 2000) (Ikonen and Kortela, 1994), have showed that the combustion process presents a highly non linear behaviour, which needs a control strategy based on nonlinear models. Therefore, a suitable model has to be developed.

Models of coal combustion for simulation purpose (Gogebakan and Selçuk, 2004), (Huilin et al., 2000), (Adanez et al., 2001), and models of other fuels, as waste combustion (Marias et al., 2001), are based on accurate considerations on combustor hydrodynamics, which are highly complex and lead to models that do not fit to be used in control strategy. In addition, most of them target combustion efficiency without considering pollutant formation.

The sludge thermal decomposition in fluidized bed is based on the reference (Werther et Ogada, 1999). The NOx formation and reduction, due to lots of chemical reactions from sludge nitrogen, are partly unknown. Only one model, dedicated to biomass combustion (Liu and Gibbs, 2002), includes many chemical reactions of NOx formation and reduction, and will be our reference for reaction scheme. The proposed model is based on the molar and energy conservation balances, and is validated with industrial data.

2. MODEL DEVELOPMENT

2.1 Sludge thermal decomposition

A typical sludge analysis is shown in Table 1.

Table 1. Sludge analysis (Werther and Ogada, 1999).

Proximate analysis	
moisture (f_{H2O} wt% raw)	76.0
char (f_{char} wt% dry)	3.6
volatiles (f_{vol} wf% dry)	45.4
ash (f_{ash} wf% dry)	51.0

After entering the furnace, sludge is decomposed into four phases under heat effect: water vapour, char, volatiles and ash, which play different roles:

- To guarantee self-combustion, the sludge produced by the wastewater treatment process sludge is dewatered mechanically to about 76% of moisture (f_{H2O}).

- Sludge char, in a solid phase that can burn, presents a very low fraction (f_{char}), unlikely carbon and biomass chars. Based on numerous laboratory analyses, J. Werther and T. Ogada (Werther and Ogada, 1999) concluded that sludge combustion behaviour is mostly governed by the gaseous phase of volatiles. So char particles can be neglected.
- Volatiles fraction (f_{vol}) is large, the volatiles combustion lead to exothermic combustion and dominate the sludge combustion process, the furnace temperature and the air input.
- Ash fraction (f_{ash}) is large.

A composition of volatiles gas, issued from off-line studies, is presented in Table 2.

Table 2. Volatiles gas composition after sludge devolatilization at temperature 760°C (Werther and Ogada, 1999).

f_{CO} wt%	43.43
f_{CO2} wt%	15.39
f_{CnHm} wt%	31.12
f_{H2} wt%	3.20
Others	6.86

The chemical species, ammonia (NH_3) and cyanide (HCN), are responsible of NOx formation, and are generally not represented in volatiles. At a relatively low combustion temperature, and with a few char but many volatiles in fuel composition, NH_3 is the dominant specie (Aho *et al.*, 1993).

2.2 Fluidized bed combustor

The fluidized sludge combustor can be divided into two beds: bubbling fluidized bed and post-combustion bed, shown in fig. 1.

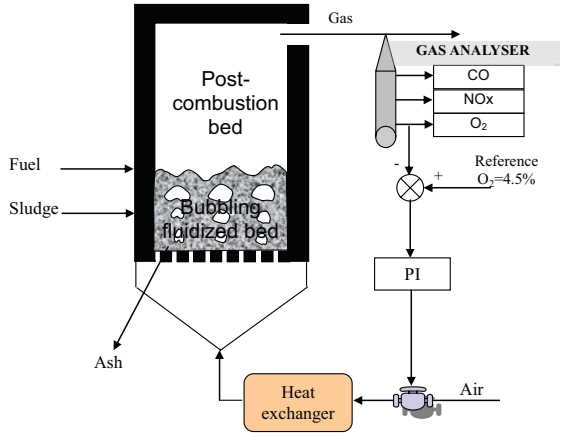


Fig. 1. Fluidized sludge combustor.

Sludge is introduced at the bottom of the furnace, i.e. the bubbling fluidized bed. Some fuel may be used as supply for starting combustion or for compensating sludge disturbances in composition or in flow. Air is preheated by a heat exchanger, and then injected with a sufficient velocity to insure the fluidization of inert sand in bubbling fluidized bed. High weight sand guarantees thermal inertia of the furnace and provides a great surface for sludge combustion and maintains a uniform temperature (~760°C). In the higher section, the post-combustion bed, volatiles gases continue post-combustion which increase the bed temperature (850°C-900°C). Temperature at the top of the furnace is limited up to 920°C, to avoid both furnace overheating and NOx formation.

The oxygen concentration is regulated up to 4.5% by air flow rate. This regulation loop provides an excess of oxygen which guaranties complete gas combustion, so as to avoid carbon monoxide (CO) formation.

On-line industrial measurements are sludge flow input in dry basis ($Q_{b,MS}^{in}$); air temperature input (T_a^{in}), air flow input (F_{am}^{in}), two bed temperatures (T^B, T^P), gas concentrations output (y_{O2} %, y_{CO} ppmv, y_{NO} mg.Nm^{-3}). As CO concentration is very low and its measurement presents lots of noise, unfortunately, this measurement can't be used for parameter estimation.

2.3 Modeling strategy and main hypothesis

The temperature and CO behaviors are described by combustion reactions. NOx formation and reduction can be considered as a separated model, using temperatures as input variables. Considering that two models has some advantages: they can be used simultaneously for simulation purpose, or separately for control purpose.

A difficulty is that sludge input flow is as constant as possible, and that sludge composition is not available on-line. So the dynamical behavior is only measured on the furnace outputs. We propose to reconstruct this composition with output measurements in the model.

As char particles can be neglected, the bubbling fluidized bed can be supposed perfectly mixed. The post-combustion bed is also approximated to a perfectly mixed reactor for model simplicity, but it is rather a plug flow reactor.

2.4 Chemical reactions

The reference (Liu and Gibbs, 2002) proposes 25 chemical reactions describing biomass combustion, including 20 reactions for NOx and N_2O formation and reduction. These reactions are selected and simplified with consideration on sludge specificities, as listed bellow:

- Due to low char content, the reactions using char as reactant and catalyst are all neglected.
- C_nH_m is supposed to be only CH_4. Its reaction is supposed to be complete in bubbling fluidized bed.
- Reaction of H_2 is supposed to be instantaneous and complete in bubbling fluidized bed.

- Catalyst as limestone (CaO), which is not used in sludge combustion, is not included.
- The lack of knowledge both on ash composition and its catalytic role (Tran et al., 2007) leads to suppose it chemically inert.
- N_2O can be decomposed rapidly in the post-combustion area where the temperature reaches 900°C. So N_2O is not considered in the model.

Five reactions are finally retained, as shown in table 3. The first reaction is NOx reduction, the second is NOx formation, and the last three are combustion reactions, which are independent from the NOx reactions. The first three reactions are supposed to be dynamically available, and the last two ones are supposed to be complete, so they do not need kinetic expression. The reactions R22, R23 and R24 are endothermic, their reaction enthalpies are noted as ΔH_{22}, ΔH_{23} and ΔH_{24}. The energy consumed or produced by the reactions R1 and R2 are negligible.

Table 3. Chemical reactions of sludge combustion model

No.	Reaction	Reaction rate r_i (mol.m^{-3}.s^{-1})	Enthalpies
R1	$NO + NH_3 + {}^1/_4O_2 \rightarrow N_2 + {}^3/_2 H_2O$	$r_1 = k_1[NH_3]^{0.5}[NO]^{0.5}[O_2]^{0.5}$	Negligible
R2	$NH_3 + {}^5/_4O_2 \rightarrow NO + {}^3/_2 H_2O$	$r_2 = k_2[NH_3][O_2]$	Negligible
R22	$CO + ½\ O_2 \rightarrow CO_2$	$r_{22} = k_{22}[CO][O_2]^{0.5}[H_2O]^{0.5}$	ΔH_{22}
R23	$CH_4 + {}^3/_2\ O_2 \rightarrow CO_2 + {}^3/_2 H_2O$	complete	ΔH_{23}
R24	$H_2 + ½\ O_2 \rightarrow H_2O$	Instantaneous	ΔH_{24}

The reaction rate r_i depends on the kinetic constant k_i and on the reactant concentrations. The kinetic constant k_i is given by the Arrhenius equation:

$$k = k_0 \exp(-\frac{Ea}{RT}) \quad (1)$$

Where k_0 is the pre-exponential factor or simply the *prefactor*, E_a is the activation energy, R is the perfect gas constant, R=8.31 J.mol^{-1}.K^{-1}, and T is the temperature (in Kelvin).

To simplify validation procedure, kinetic constant k_i is supposed to be constant in one bed, so the rate constant of reaction j in bubbling fluidized bed and post-combustion bed are named as k_j^B and k_j^P.

2.5 Combustion modelling

Combustion reactions are mostly responsible of temperature and oxygen concentration in the furnace. As the combustion reactions are independent of the NOx reactions, a model of combustion can be proposed individually.

After entering the furnace, sludge particles are decomposed physically and chemically at high temperature. Firstly, drying and devolatilization take place simultaneously, splitting sludge into species described previously in table 1 and table 2. Flux of gas species can be reconstructed:

$$\begin{bmatrix} F_{H_2O}^{in} \\ F_{CO}^{in} \\ F_{CH_4}^{in} \\ F_{H_2}^{in} \end{bmatrix} = \begin{bmatrix} Q_{b,MS}^{in} \dfrac{f_{H_2O}}{1-f_{H_2O}} / M_{H_2O} \\ Q_{b,MS}^{in} . f_{vol} f_{CO} / M_{CO} \\ Q_{b,MS}^{in} . f_{vol} f_{CH_4} / M_{CH_4} \\ Q_{b,MS}^{in} . f_{vol} f_{H_2} / M_{H_2} \end{bmatrix} \quad (2)$$

Where F_i^{in} is the molar flow rate of component i after sludge drying and devolatilization, mol.h^{-1}; M_i is the molecular weight of component i, kg.mol^{-1}.

The volatile gas repartition is taken from table 2 without adaptation, though some relations between species fractions and temperature and fuel composition should be more realistic. With this assumption, only two sludge characteristics are needed: water fraction (f_{H2O}) and volatile fraction (f_{vol}). They are calculated by two static global balances: one is the balance of oxygen; the other is the balance of thermal energy, as shown in fig. 2.

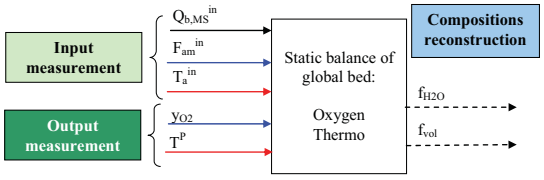


Fig. 2. Reconstruction f_{H2O} and f_{vol} from measurements (dashed line: unmeasured data, solid line: measured data).

Figure 3 presents the model structure. Three macroscopic inputs that can be easily measured (solid line): sludge flow rate in dry basis ($Q_{b,MS}^{in}$), air flow rate (F_{am}^{in}), input air temperature (T_a^{in}). State variables are gaseous species concentrations (C_{H2O}^B, C_{CO}^B, C_{O2}^B and C_{H2O}^P, C_{CO}^P, C_{O2}^P) and bed temperatures (T^B and T^P). Only four variables are measured (solid lines: C_{CO}^P, C_{O2}^P, T^B and T^P).

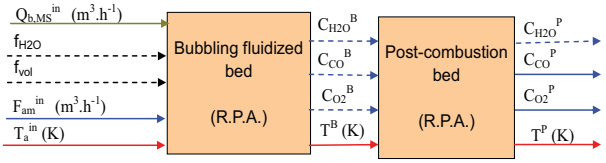


Fig. 3. Structure of combustion model (dashed line: unmeasured data, solid line: measured data).

The combustion model is written by molar and energy conservation balances (^{B} bubbling bed, ^{P} post-combustion):

$$\begin{pmatrix} \dfrac{dC_{H_2O}^B}{dt} \\ \dfrac{dC_{CO}^B}{dt} \\ \dfrac{dC_{O_2}^B}{dt} \end{pmatrix} = \begin{bmatrix} F_{H_2O}^{in} \\ F_{CO}^{in} \\ F_{O_2}^{in} \end{bmatrix} / V_B - \begin{pmatrix} C_{H_2O}^B \\ C_{CO}^B \\ C_{O_2}^B \end{pmatrix} \times \frac{F_g^B}{V^B} + \begin{pmatrix} \varphi_{H_2O}^B \\ \varphi_{CO}^B \\ \varphi_{O_2}^B \end{pmatrix} \quad (3)$$

$$\begin{pmatrix}\varphi^B_{H_2O}\\ \varphi^B_{CO}\\ \varphi^B_{O_2}\end{pmatrix}=\begin{bmatrix}0\\ -1\\ -0.5\end{bmatrix}\times r^B_{22}+\begin{pmatrix}F^{in}_{H_2}+2F^{in}_{CH_4}\\ F^{in}_{CH_4}\\ -0.5F^{in}_{H_2}-1.5F^{in}_{CH_4}\end{pmatrix}/V_B \quad (4)$$

Where F_{O2}^{in} is the oxygen molar flow rate brought by input air flow, mol.h^{-1}; F_g^{in} is the global gas flow rate (calculated by a global static balance on input and output flows), m^3.h^{-1}; V_B is the bubbling bed volume, m^3; φ_i^B is the production or consumption flux of component i by chemical reactions; r_{22}^B is the reaction rate of R22 (see table 3).

$$\begin{aligned}c_{ps}m_s^B\frac{dT^B}{dt}=&\left(c_{pa}F^{in}_{am}\rho_{a(T^{in}_{am})}T^{in}_a+c_{pb}\frac{Q^{in}_{b,MS}}{1-f_{H_2O}}T^{in}_b\right)\\&-\left(c_{pg}\rho_{g(T^B)}F^B_gT^B+c_{p,ash}Q^{in}_{b,MS}(1-f_{vol})T^B\right)\\&-\left(F^{in}_{H_2O}L_{H_2O}\right)-\left(r^B_{22}\Delta H_{R22}V^B+F^{in}_{CH_4}\Delta H_{R23}+F^{in}_{H_2}\Delta H_{R24}\right)\end{aligned} \quad (5)$$

Where c_{ps}, c_{pa}, c_{pg} and $c_{p,ash}$ are specific heat capacities of sand, air, gas and ash, J.kg^{-3}.K^{-1}; m_s^B is the sand mass in bubbling fluidized bed, kg; ρ is the density, kg.m^{-3}; L_{H2O} is the water latent heat of vaporisation, J.mol^{-1}.

Post-combustion balances can be similarly written:

$$\begin{pmatrix}\frac{dC^P_{H_2O}}{dt}\\ \frac{dC^P_{CO}}{dt}\\ \frac{dC^P_{O_2}}{dt}\end{pmatrix}=\begin{pmatrix}C^B_{H_2O}F^B_g-C^P_{H_2O}F^P_g\\ C^B_{CO}F^B_g-C^P_{CO}F^P_g\\ C^B_{O_2}F^B_g-C^P_{O_2}F^P_g\end{pmatrix}/V^P+\begin{pmatrix}\varphi^P_{H_2O}\\ \varphi^P_{CO}\\ \varphi^P_{O_2}\end{pmatrix} \quad (6)$$

$$\begin{pmatrix}\varphi^P_{H_2O}\\ \varphi^P_{CO}\end{pmatrix}=\begin{bmatrix}0\\ -1\end{bmatrix}\times r^P_{22} \quad (7)$$

$$\begin{aligned}c_{pg}m_g^P\frac{dT^P}{dt}=&\left(c_{pg}\rho_{g(T^B)}F^B_gT^B-c_{pg}\rho_{g(T^P)}F^P_gT^P\right)\\&-\left(r^P_{22}\Delta H_{R22}V^P\right)\end{aligned} \quad (8)$$

Only kinetic parameters (k_{22}^B and k_{22}^P) need to be estimated, other parameters are known from literature.

2.6 NOx modelling

As the combustion model has modelled the bed temperatures and the oxygen concentrations, which can be used as inputs of the NOx model. They are provided either from the combustion model or directly by measurements.

The NOx model can be then established and simulated independently of the combustion model, as shown in fig.4.

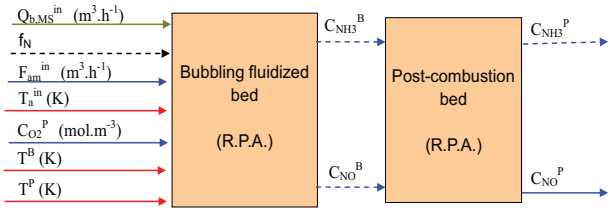


Fig. 4. Structure of NOx model.

One input cannot be measured on-line: the sludge nitrogen content (f_N), it is supposed to change slightly, and may be an additional parameter to be estimated. State variables are gaseous nitrogen species (C_{NH3}^B, C_{NO}^B, C_{NH3}^P and C_{NO}^P) among which only one is measured (C_{NO}^P).

After devolatilization, flux of NH_3 is reconstructed:

$$F^{in}_{NH_3}=Q^{in}_{b,MS}f_{vol}f_N/M_N \quad (9)$$

The balance equations are:

$$\begin{pmatrix}\frac{dC^B_{NH_3}}{dt}\\ \frac{dC^B_{NO}}{dt}\end{pmatrix}=\begin{pmatrix}F^{in}_{NH_3}\\ 0\end{pmatrix}/V_B-\begin{pmatrix}C^B_{NH_3}\\ C^B_{NO}\end{pmatrix}\times\frac{F^B_g}{V^B}+\begin{pmatrix}\varphi^B_{NH_3}\\ \varphi^B_{NO}\end{pmatrix} \quad (10)$$

$$\begin{pmatrix}\varphi^B_{NH_3}\\ \varphi^B_{NO}\end{pmatrix}=\begin{bmatrix}-1&-1\\ -1&1\end{bmatrix}\times\begin{pmatrix}r^B_1\\ r^B_2\end{pmatrix} \quad (11)$$

$$\begin{pmatrix}\frac{dC^P_{NH_3}}{dt}\\ \frac{dC^P_{NO}}{dt}\end{pmatrix}=\begin{pmatrix}C^B_{NH_3}F^B_g-C^P_{NH_3}F^P_g\\ C^B_{NO}F^B_g-C^P_{NO}F^P_g\end{pmatrix}/V^P+\begin{pmatrix}\varphi^P_{NH_3}\\ \varphi^P_{NO}\end{pmatrix} \quad (12)$$

$$\begin{pmatrix}\varphi^P_{NH_3}\\ \varphi^P_{NO}\end{pmatrix}=\begin{bmatrix}-1&-1\\ -1&1\end{bmatrix}\times\begin{pmatrix}r^P_1\\ r^P_2\end{pmatrix} \quad (13)$$

Only kinetic parameters (k_1^B, k_1^P, k_2^B and k_2^P) need to be estimated, other parameters are known from literature.

3. VALIDATION

3.1 Validation strategy

Because of bad numerical results, it is not possible to simulate the model with the parameters at their literature values. The proposed validation strategy is based on some physical considerations and sensitivity analysis. As the number of parameters is numerous, only some kinetic parameters are estimated. This choice is made on the consideration that they influence directly the dynamic behaviour of the model, and that the literature values were proposed from chemical engineering laboratory analysis of coal combustion, which means that the identified values may very different.

Another important choice was to identify the parameters with two separated models or with a global one. In the first step, we choose the first solution considering that NOx content can't influence the furnace temperature. In addition, temperature is difficult to be estimated accurately with a model, though its measurement is quite reliable. Some attempts for identification with the global model have not improved the results. The main drawback of validation is that part of sludge composition (f_N, f_H) has to be fixed for each data set. Now it is adjusted by trial and error, but it would have to be further included in global parameters.

3.2 Industrial data file for parameter estimation

Two industrial data files of an industrial fluidized sludge combustor with different events are used for parameters estimation and model validation. The data used for identification corresponds to the introduction of fat matters in furnace. Model input measurements are shown in fig. 5.

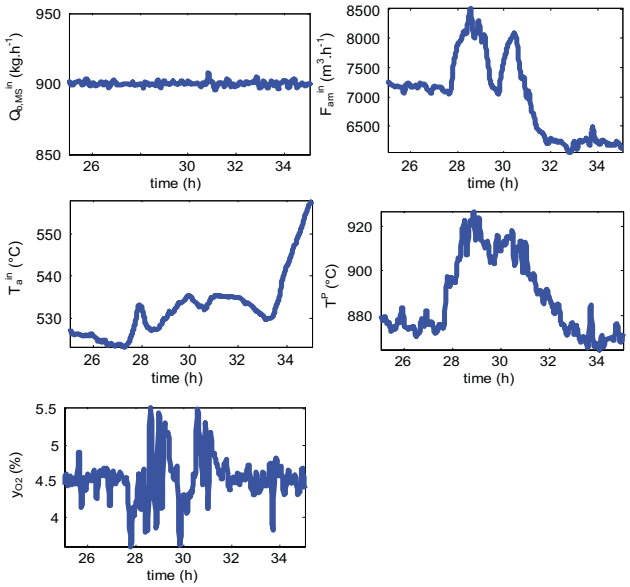


Fig. 5. Measurements of model input of data file 1.

Sludge flow ($Q_{b,MS}^{in}$) in Fig. 5 is constant, which is a necessity as sludge has to be treated continuously. The dynamic event is revealed by output measurements, which affect the input by the heat exchanger (T_a^{in} is defined by T^P) and the oxygen control loop (F_{am}^{in} is the action which regulates y_{O2}). The post-combustion temperature presents an increasement due to the dynamic event. The oxygen concentration (y_{O2}) is regulated at reference point 4.5%.

3.3 Parameter estimation methodology

For parameter estimation, minimisation of a least square criterion on the difference between the measured value and the model value has been carried out.

For combustion model, only temperature to be used is T^B, as T^P is used to reconstruct the sludge compositions f_{H2O} and f_{vol}. Consequently, the parameter k_{22}^P, can't be estimated. Some additional measurements, such as y_{CO}, may be useful for estimating this parameter. In the model, k_{22}^P is taken as a great value to guarantee a complete CO combustion, because CO in the measurement is almost null in the sake of oxygen regulation.

For NOx model, four kinetic parameters k_1^B, k_2^B, k_1^P and k_2^P need to be identified. Their literature values, calculated from steady temperature values (T^B=752°C, T^P=870°C) are presented in table 4.

Table 4. Literature values of parameters to be estimated

$k_1^B = 0.4$	$k_1^P = 7.2$
$k_2^B = 0.02$	$k_2^P = 0.87$

Because of the great number of parameters to be estimated with only one measured data y_{NO}, a sensitivity analysis is made to select the more influent parameter. Figure 6 shows that k_1^P is the most sensitive parameter with respect to y_{NO}. Finally, only the parameter k_1^P is estimated as a constant, the others are used with their reference values.

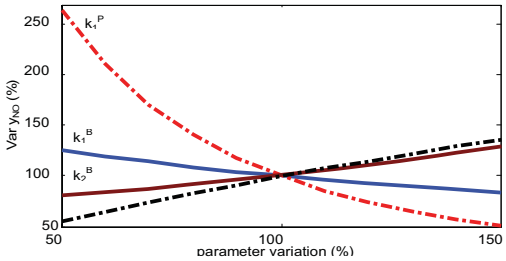


Fig. 6. Sensitivity analysis of k_1^B, k_2^B, k_1^P and k_2^P with respect to y_{NO}.

3.4 Estimated Parameters

The initial value of k_{22}^B=$3.25\times10^7\exp(-15098/T^B)$ (14)

Which lead the equation (15) after estimation:

k_{22}^B=$(2211\pm5)\exp(-(7743\pm2)/T^B)$ (15)

The estimated value of k_1^P:

k_1^P=0.9981 ± 0.0035 (16)

The estimated values are all very small compared to the literature values. These differences can be attributed to the great simplifications of chemical reactions: neglected char, instantaneous and complete reactions of H_2 and CH_4 in the bubbling bed, which need to slow the global combustion reactions. For post-combustion, perfectly mixed reactor may also be revised. Some tests with other initial values have leading the same results. However, the precision interval points out that no better values can be estimated.

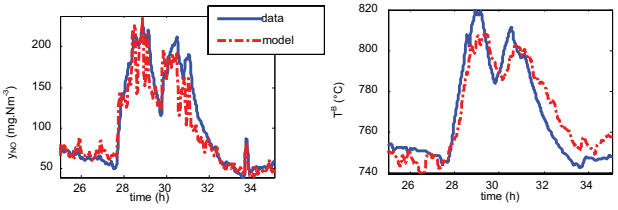


Fig. 7. Comparison of data file and model with kinetic parameter estimated k_{22}^B and k_1^P.

With the estimated values in equation (15) and (16), simulation results are shown in fig. 7. Bubbling fluidized bed temperature (T^B) in combustion model is tracking well the measurement, and NOx model fits also very well the measured data. In conclusion, we can consider that the model

is realistic and these results are good enough to continue a validation step.

3.5 Validation with data file 2

The main characteristic of data file 2 (fig. 8) is an increase of the operating point: NOx concentration exceeds the norm of 400 mg.Nm^{-3}, post-combustion bed temperature (T^P) is upper the limitation value of 920°C. The sludge flow rate ($Q_{b,MS}^{in}$) remains constant at 900 kg.h^{-1} and is not represented. The air flow rate (F_{am}^{in}) and the oxygen concentration present small fluctuations around a reference point.

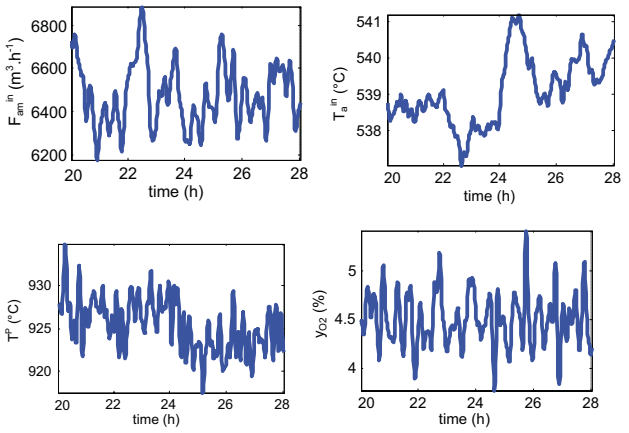


Fig. 8. Measurements of model input of data file 2

When analysing the measurements, we can notice that the bed temperatures increase greatly but fewer oxygen is consumed due to perturbation reconstruction of water fraction and volatile fraction. Trying to translate this behaviour in the model, the following adjustments are made: the hydrogen fraction (f_{H2}) after devolatilization is decreased from 3.2% to 2% (see table 2), the nitrogen fraction (f_N) is increased from 1% to 2.5%. Both corrections are used to fill up the unknown perturbations.

Figure 9 compares the simulation results with the measurements. We can see that the global tendency is good, and that means that our model is able to fit the data. The fluctuations observed for the model are due to algebraic equations for the reconstruction of f_{H2O} and f_{vol}. The main drawback is that the lack of knowledge on the input composition is a major obstacle to such modelling.

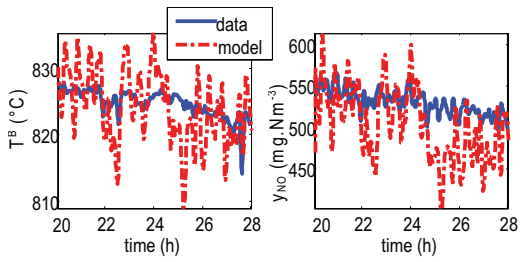


Fig. 9. Model validation.

4. CONCLUSIONS

A dynamic model of fluidized bed sludge combustor has been designed to predict NOx emissions. The main hypothesis is the lack of reactant particles in the furnace, leading to only five chemical reactions and simplifying hydrodynamics into two perfectly mixed reactors. As the sludge composition is not available, it has been reconstructed with output measurements. This point represents the main drawbacks of the model: more knowledge on devolatilization and/or more output measurements would be very helpful to have a deterministic model. However this model has been partly validated with industrial data files, and it is sufficiently representing the NOx formation behaviour, to be used in a control strategy and therefore to contribute to improve combustion quality and control NOx emissions.

REFERENCES

Adanez J., Gayán P., Grasa G., Diego L.F., Armesto L., Cabanillas A., (2001), Circulating fluidized bed combustion in the turbulent regime: modelling of carbon combustion efficiency and sulphur retention, *Fuel*, 80, 1405-1414.

Aho M.J., Hämäläinen J.P., Tummavuor J.L., (1993), Importance of solid-fuel properties to nitrogen-oxide formation through HCN and NH3 in small-particle combustion. *Combustion Flame*, 95, 22-30.

Bittanti S., Bolzern P., Campi M.C., Marco A., Poncia G., Prandoni W., (2000), A model of a bubbling fluidized bed combustor oriented to char mass estimation, *IEEE transactions on control systems technology*, 8 (2).

Gogebakan Y., Selçuk N., (2004), Assessment of a model with char attrition for a bubbling atmospheric fluidized-bed combustor, *Combust. Sci. and Tech.*, 176, 799-818.

Huilin L., Guangbo Z., Rushan B., Yongjin C., Gidaspow D., (2000), A coal combustion model for circulating fluidized bed boilers, *Fuel*, 79, 165-172.

Ikonen E. and Kortela U., (1994), Dynamic model for a bubbling fluidized bed coal combustor, *Control Eng. Practice*, 2 (6), 1001-1006.

Liu H., Gibbs B.M., (2002), Modeling of NO and N2O emissions from biomass-fired circulating fluidized bed combustors, *Fuel*, 81, 271-280.

Marias, F., Puiggali, J.R., Flamant, G., (2001), Modelling for Simulation of Fluidized-Bed Incineration Process, *AIChE Journal*, 47, 1438-1460.

Muir J.R., Brereton C., Grace J.R. and Lim C.J., (1997), Dynamic modeling for simulation and control of a circulating fluidized bed combustor, *AIChE Journal*, 43 (5).

Reimann D.O., (1999), Problems about sludge incineration, *Proceedings of the Workshop on "Problems around sludge", session 3: Technology and innovative options related to sludge management*, Stresa, Italy, 3, 173-183.

Tran K.Q., Kilpinen P., Kumar N., (2007), In-situ catalytic abatement of NOx during fluidized bed combustion – a literature study, *Applied Catalysis B: Environmental*, 2007.

Werther J., Ogada T., Sewage sludge combustion, *Progress in energy and combustion science*, 25, pp. 55-116, 1999.

Comparison of Different Modeling Concepts for Drying Process of Baker's Yeast

U. Yüzgeç* M. Türker **

**Kocaeli University Department of Electronics & Telecom. Engineering, 41040 Kocaeli, Turkey*
(uyuzgec@kocaeli.edu.tr)
***Pakmaya, P.O. Box 149, 41001, Kocaeli, Turkey*
(mustafat@pakmaya.com.tr)

Abstract: This study investigates different modeling approaches and compares for drying of baker's yeast in a fluidized bed dryer. Four modeling concepts were investigated: modeling based on the mass and energy balance, modeling based on diffusion mechanism in the granule, modeling based on recurrent Artificial Neural Network (ANN) and modeling based on Adaptive Neural Network Fuzzy Inference System (ANFIS). Dry matter of product, product temperature and product quality were predicted using these model structures. To evaluate performances of the modeling structures, industrial scale drying process data were used.

Keywords: Drying process, spatial distribution, product quality, modeling, ANN, ANFIS

1. INTRODUCTION

Biological products, such as agricultural products, foods, pharmaceuticals, enzyme preparations, bacterial and yeast cultures, are particularly sensitive to drying conditions, including moisture content and temperature (Yüzgeç et al., 2008). There are different modeling concepts for the drying processes reported in the literature. In general, three possible approaches can be taken to the modeling: a physical approach based on energy and mass balance (Temple et al., 2000; Türker et al., 2006), black-box modeling (Castellanos et al., 2002; Köni et al., 2009) and hybrid modeling (Ciesielski et al., 2001). In contrast with the physical approach, the black-box modeling approach does not require prior theoretical knowledge.

The objective of this study is to examine alternative and novel modeling techniques based on the physical approaches and based on mathematical approaches, such as ANN and ANFIS structures, for an industrial-scale drying process of baker's yeast. Four model structures which are different from each other were investigated. First model was developed by using mass and energy balances in the fluid-bed. The second mathematical model is based on the spatial distribution of moisture, temperature and quality. In the third model, the recurrent ANN structure was selected according to the results of the regression analysis in their study presented by Köni et al.(2009). The last model is constructed using ANFIS architecture. The vast numbers of industrial data (570 data sets) used for training and testing of the models were collected from a production-scale baker's yeast drying process. As a result of this work, advantages and disadvantages of the model structures investigated for drying process were presented.

2. MATHEMATICAL MODELS

2.1 Model based on mass and energy balances

Model equations are constructed by combining mass and energy balances (Temple and Boxtel, 1999; Kanarya, 2002).

The model equations consist of four basic balance equations: Dry solid balance, water conservation, air conservation and energy balance. All of these equations are presented below:

$$\frac{dM_{b,y}}{dt} = m_y^i - m_y^o \tag{1}$$

$$\frac{dW_{b,y}}{dt} = w_y^i - r_w - w_y^o \tag{2}$$

$$\frac{dM_{b,a}}{dt} = m_a^i - m_a^o \cong 0 \tag{3}$$

$M_{b,y}$ is dry solid mass of product in the bed and m_y^i is the flow rate of the product into the bed and m_y^o is the flow rate of the product out of the bed. $W_{b,y}$ represents the water mass inside the bed, w_y^i is the flow rate of water in product fed to the system, r_w is the flow rate of water removed from product by means of evaporation, and w_y^o represents the flow rate of water in product entrained through cyclones. $M_{b,a}$ is the mass of air in the bed, m_a^i, m_a^o represent the flow rates of the inlet and outlet air.

The energy accumulation of the process H is assumed as adiabatic. The energy accumulation in the bed can be written as dynamic balance between energy flows to and from the system as given in Eq.(4):

$$H = h_y^i + h_a^i - h_y^o - h_a^o. \tag{4}$$

In this equation, h_a^i represents the energy introduced by air, h_y^i is the energy introduced by yeast, h_a^o is the energy removed by air and h_y^o represents the energy removed by yeast. The detail informations related to this model can be found in the paper presented by Türker et al. (2006).

2.2 Model based on spatial distributions of moisture and quality

This modeling of the drying comprises four main parts: moisture diffusion equation, heat balance equation, shrinking model, product activity in the granule. The model also includes dependence of the moisture and temperature of granules on several parameters like moisture diffusion coefficient, heat and mass transfer coefficients and water activity. The batch fluidized bed is assumed to be an ideally mixed bed, with uniform temperature and humidity of the air which equal the outgoing air conditions. The particles are all at the same stage of drying at any instant of the batch operation. Furthermore, there is no interaction between the particles, as far as drying is concerned (Yüzgeç et al., 2008).

A generalized formulation of the moisture diffusion equation is presented by a nonlinear partial differential equation (Schoeber, 1976) as given below:

$$\frac{\partial(\rho_s X)}{\partial t} = \frac{1}{r^v}\frac{\partial}{\partial r}\left(r^v \rho_s D(X,T)\frac{\partial X}{\partial r}\right). \tag{5}$$

X (*kg water/kg dry solid*) is the moisture content inside the granule, *D* (m^2/s) is the moisture diffusion coefficient which is the function of material's moisture content (*X*) and temperature (*T*) and *v* represents geometry factor with *v* = 0 slab, *v* = 1 cylinder, *v* = 2 sphere. The initial and boundary conditions:

$$t = 0; 0 \le r \le R_d \Rightarrow X(0,r) = X_0 \tag{6}$$

$$t > 0 \Rightarrow \left.\frac{\partial X}{\partial r}\right|_{r=0} = 0 \tag{7}$$

$$t > 0 \quad r = R_d \Rightarrow j_{m,i} = -D\rho_s \left.\frac{\partial X}{\partial r}\right|_{r=R_d} = k\left(\rho_{wv,i} - \rho_{wv,g}\right) \tag{8}$$

$j_{m,i}$ is the moisture flux at the interface, *k* is the liquid film mass transfer coefficient around the granule, $\rho_{wv,i}$ represents the water vapor concentration at the interface and $\rho_{wv,g}$ represents the water vapor concentration in the bulk air.

The heat balance can be described as heat transfer both to and from the surface and within the material. The equation of the heat balance inside a granule is described by the following non-linear partial differential equation (Quirijns et al., 1998; Quirijns et al., 2000),

$$\frac{\partial(T(\rho_s c_{p,s} + \rho_m c_{p,m}))}{\partial t} = \frac{1}{r^v}\frac{\partial}{\partial r}\left(r^v \lambda \frac{\partial T}{\partial r}\right) \tag{9}$$

where *T* is the temperature, ρ_m is the moisture concentration, ρ_s is the dry solid concentration inside the granule, $c_{p,s}$ and $c_{p,m}$ are the heat capacities of the solid and moisture, λ is the thermal conductivity of the granule. The initial and boundary condition are given by

$$t = 0; 0 \le r \le R_d \Rightarrow T(0,r) = T_0 \tag{10}$$

$$t > 0 \Rightarrow \left.\frac{\partial T}{\partial r}\right|_{r=0} = 0 \tag{11}$$

$$t > 0 \quad r = R_d \Rightarrow j_{T,i} = -\lambda \left.\frac{\partial T}{\partial r}\right|_{r=R_d} \tag{12}$$

$$j_{T,i} = \alpha(T(t,R_d) - T_a) + \Delta H_v\big|_{T(t,R_d)} j_{m,i} \tag{13}$$

where $j_{T,i}$ is the heat flux at the interface, α is the heat transfer coefficient, T_a is the inlet air temperature and ΔH_v is the evaporation enthalpy of water.

The product quality can be described as first-order kinetics (Lievense, 1991):

$$\frac{dQ}{dt} = -k_e Q \tag{14}$$

where *Q* is the concentration of the active product and k_e is the specific rate of product activity. According to the Arrhenius equation, the rate of the product activity can be expressed as a function of the temperature (Liou et al., 1984). Lievense (1991) has described that following equation for dependency of $\ln(k_e)$ on temperature and moisture:

$$\begin{aligned}\ln(k_e) = &\left[\left(a_1 - \frac{a_2}{RT}\right)X + \left(b_1 - \frac{b_2}{RT}\right)\right] \\ &+ \left[1 - \exp\left(pX^q\right)\right]\left[\left(a_3 - \frac{a_4}{RT}\right)X + \left(b_3 - \frac{b_4}{RT}\right)\right]\end{aligned} \tag{15}$$

where *p, q, a_i, b_i* are the parameter values in the equation. If *p<0* and $q \ge 1$, at high moisture content, $\exp\left(pX^q\right) \approx 0$ and $\ln(k_e)$ consists of the linear sum of the two parts; at low moisture content, $\exp\left(pX^q\right) \approx 1$ and $\ln(k_e)$ is described with the first linear part of the equation.

The volume of the granule consists of volumes of both moisture and solid:

$$V = V_m + V_s \tag{16}$$

According to Coumans (1987), the volume of the granule is a linear function of the average moisture content $\overline{X}$ during shrinkage that is expressed by

$$V = V_s\left(1 + \tau \frac{d_s}{d_m}\overline{X}\right) \tag{17}$$

where τ is the shrinkage coefficient within $0 \le \tau \le 1$. The detail informations related to this model can be found in the paper presented by (Yüzgeç et al., 2008).

2.3 Model based on Artificial Neural Network (ANN)

The fifteen different ANN structures were investigated in the study introduced by Köni et al. (2009) in order to determine the suitable model which represents drying process. The essential differences for all of the proposed ANN structures are in the input layer and in the relations of the hidden layers. The ANN-9 model had the best performance according to the results of regression analysis of all model approaches. Fig.1 presents the architecture of this ANN model. This recurrent neural network model has nine layers each of whose hidden neurons are twelve and it consists of five inputs and three outputs (*DM* is dry matter of product (%), T_m is product temperature (^oC) and ΔDM is change in dry matter of product).

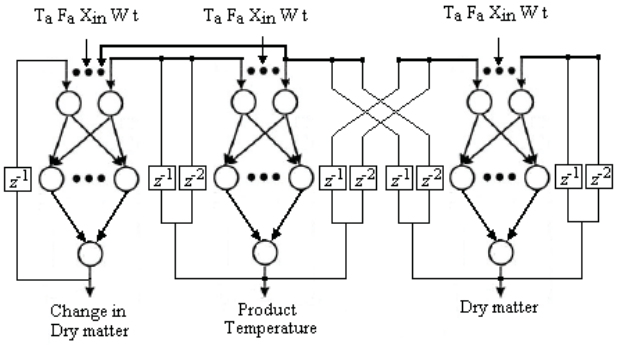


Fig.1. Drying model architecture designed using recurrent ANN. *t*: the drying time (s), *W*: loading (kg), X_{in}: moisture content of inlet air (kg water/kg air), F_a: flow rate of inlet air (m^3/h) and T_a: temperature of inlet air (^oC).

The drying process should be performed under optimal conditions in order to minimize quality loss. The change in the quality loss (ΔQ) is given as:

$$\Delta Q = \frac{Q_n - Q_f}{Q_n} \tag{18}$$

where Q_n is the quality at the beginning of the drying and Q_f is final quality at the end of the drying. In the industrial scale production, product quality is measured at the beginning and at the end of the drying process. In this study, a neural network model with three layers was used as quality model based on the results of regression analyses done by Köni et al. (2009). In this model shown in Fig.2, the inputs were considered as the process output variables, such as dry matter of product (*DM*) and product temperature (T_m). The values of dry matter and product temperature were stored in each drying time and a database forms for the product quality obtained at the end of the process. The inputs of quality model are all sampled values of dry matter of product *DM(i)* and product temperature $T_m(i)$, i denotes the sample in a drying period (Köni et al., 2009).

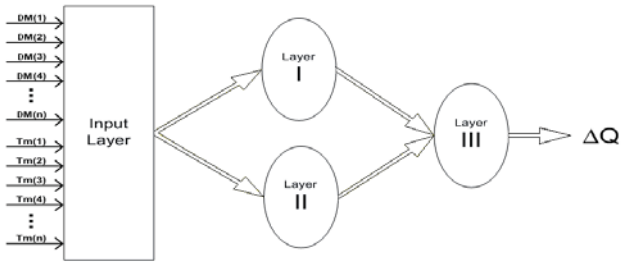


Fig. 2. Architecture of ANN model for quality.

The first layer represents the quality effect of drying on baker's yeast and the second layer denotes the effect arising from fermentation process. The tansig functions in layer 1 and layer 3 and logsig function in layer 2 are used as activation functions.

2.4 Model based on Adaptive Neural Network-Based Fuzzy Inference System (ANFIS)

The basic difference between ANFIS and ANN architecture is that ANFIS has a single output. This means that different ANFIS structures are constructed for each output parameter to be predicted, without changing the input parameters. Five input parameters were applied to the proposed ANFIS model approach: drying time, loading weight, moisture content of inlet air, flow rate of inlet air and temperature of inlet air. ANFIS structures were constructed separately for fuzzy modeling to predict the dry matter of the product (*DM*), the product temperature (T_m) and the change in dry matter of the product (ΔDM). In Fig. 3, the ANFIS architecture proposed for the dry matter of the product is given in detail. A_i, B_i, C_i, D_i and E_i (i = 1,2,3) represent the drying time (t), loading weight (W), moisture content of inlet air (X_{in}), flow rate of inlet air (F_a) and temperature of inlet air (T_a) membership functions, respectively.

All of the outputs are predicted by linear output functions. The ANFIS structures are the same for the other two output parameters, namely, *Tm* and *ΔDM*. Only output parameters change, while the inputs are kept the same. In this model, all of the ANFIS structures have five inputs and a single output, using the Sugeno-type fuzzy model. Three membership functions are defined for each input parameter. Although the model causes overloads during operation, all output membership functions were defined as first-order (linear). In the neuro-fuzzy model, the parameters associated with each membership function were adjusted by a hybrid learning algorithm consisting of a combination of least-squares and back propagation gradient descent methods. This algorithm used back propagation for the parameters related to the input membership functions and least squares estimation for the parameters related to the output membership functions (Jang, 1993; Azeem et al., 2000). As a result of using the hybrid learning algorithm, the training error decreased during the learning process.

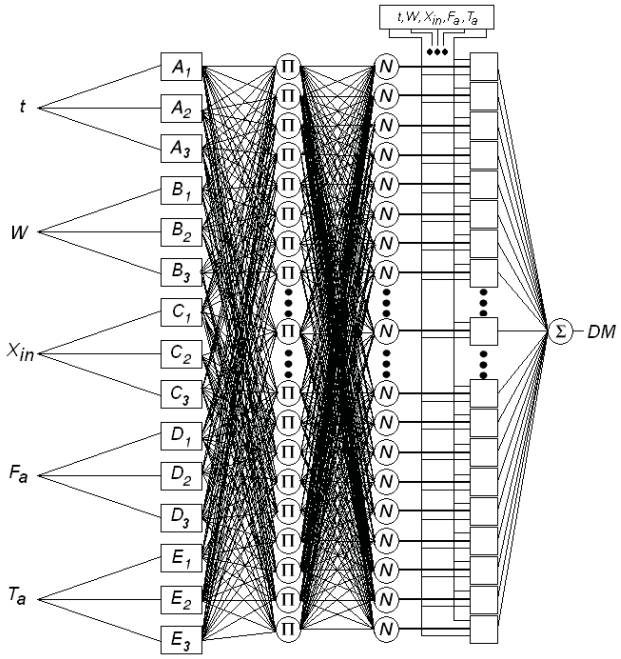


Fig. 3. Inputs and output of fuzzy model structure for dry matter of product (*DM*).

3. MATERIALS & METHODS

Data obtained from an industrial scale drying plant, which is produced baker's yeast (*Saccharomyces cerevisiae*), were used to test the models in this study. Yeast cake was extruded into the dryer through a perforated plate of a different diameter to obtain the desired granule size (Türker et al., 2006). In general, baker's yeast with a value of 33-34% dry matter prior to loading in the dryer eventually dried to a value of 94-96% dry matter. The fluid bed contained a centrifugal fan to supply air drawn from the ambient air. There are two essential output parameters for the drying processes: moisture content and product temperature. The product temperature was measured by Pt-100 sensors in the fluid bed. The temperature on the dryer outlet was also measured regularly. Moisture content is more difficult to measure than temperature. Infrared sensors are used to measure the moisture content in the drying material at third drying stage. The data set consisted of measurements obtained from the dryer under different loading conditions and different air profiles over one year of training and testing for the ANN and ANFIS model approaches. The sampling period of the data collection was 30 seconds.

The quality of the product was defined as the volume of carbon dioxide produced per unit time upon introduction of the yeast into the dough. This method is commonly used in the yeast industry to assess the performance of baker's yeast (Yüzgeç et al., 2008). To measure product quality, yeast samples are taken from the dryer at specific times during the drying, and then the volume of carbon dioxide in the laboratory is measured. Relative activity is expressed as the ratio of the activity of the product at time t to the activity of the yeast cake at the beginning time of the drying. A database which consists of 570 data was divided into the two parts: 60% training and 40% testing (Köni et al., 2009).

The first mathematical model is based on the first order ordinary differential equation with initial and boundary conditions. Therefore Runge-Kutta finite difference method has been used for the solution of the equation describing product temperature (Türker et al., 2006). There are two nonlinear partial differential equations in the second mathematical model. Due to the complex nature of the analytical solutions, numerical methods especially Crank-Nicholson method were used for the solution of non-linear partial temperature and moisture diffusion (Yüzgeç et al., 2008). The equations were subject to a Robin boundary condition. The mass flux at the interface is variable in the Robin type boundary condition. The sample time was chosen as one second and the particle was divided into ten grids.

4. RESULTS & DISCUSSION

To evaluate the performances of developed model structures, outputs of the models (dry matter of product and product temperature) were compared with the industrial scale data obtained from drying process of baker's yeast. Fig.4 show the simulation results for the energy and mass balance based model and the spatial distributions of moisture and quality based model.

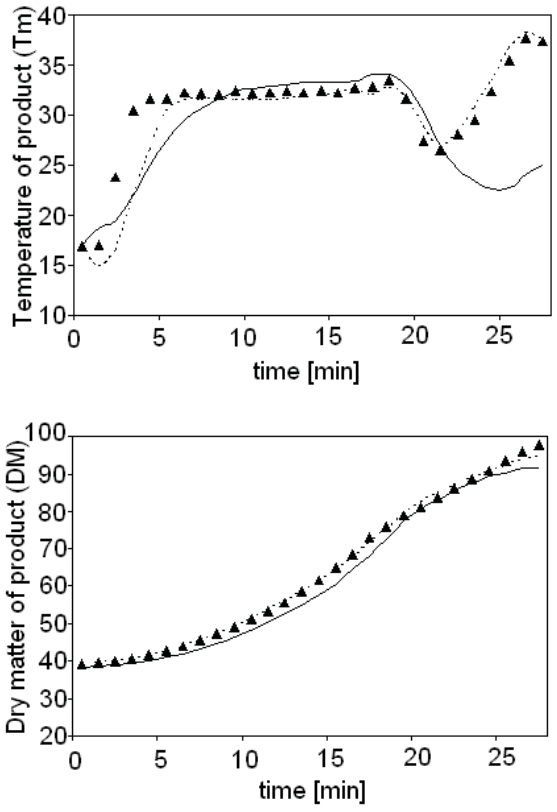


Fig. 4. Simulation results of the energy and mass balance based model (—) and granule based model (---). Industrial data (▲).

Note from this figure that there is good correspondence between the results obtained by developed model approaches and experimental data. Compared to the energy and mass balance based model, the granule based model significantly improves the predictions during the drying of granular product with spatial distribution of moisture. The simulation result of ANN based model is shown in Fig.5, together with experimental data chosen randomly from the database. The features of this experimental data is: loading weight 550-650 kg, moisture content of inlet air 0.0037-0.0056 kg water/kg air, flow rate of inlet air 34000-47500 m^3/h and temperature of inlet air 54-132 ^0C. In these figures, the predicted values for the drying process match very well with the experimental data; the differences can only be seen on a much finer scale. Comparison between the simulation results obtained by ANFIS based model and experimental data set is presented in Fig. 6. It can be seen from these figures that the ANFIS based model approach has a little more performance than that of the ANN based model. There is no vital difference between the results of the last two model structures. Both of them can be used as alternatives to one another with respect to response time and system definition. ANFIS based models have more advantages due to their adaptive structure.

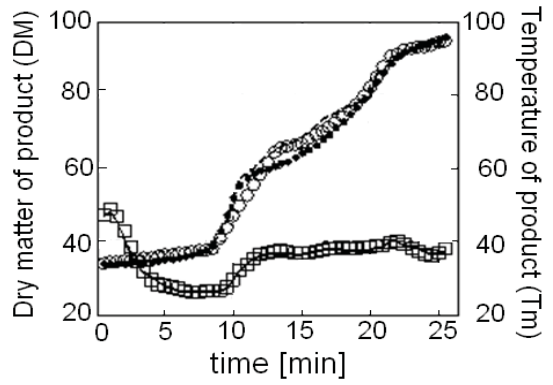


Fig. 5. The simulation results of the ANN based model. *DM* experimental, (o) *DM* simulation (---), *ΔDM* simulation (•••), T_m experimental (□),T_m simulation (—).

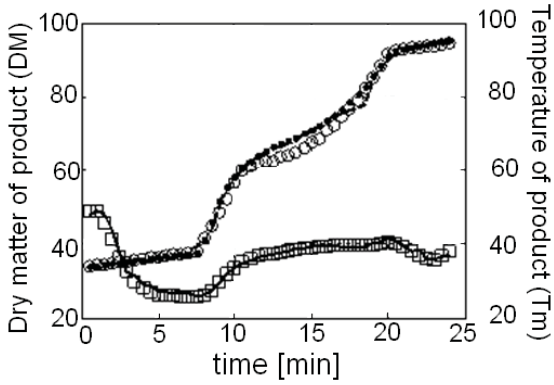


Fig. 6. The simulation results of the ANFIS based model. *DM* experimental, (o) *DM* simulation (---), *ΔDM* simulation (•••), T_m experimental (□),T_m simulation (—).

For modeling of the product quality loss or product activity using ANN and ANFIS approaches, which is the most difficult variable in the design of a proper physical model for biomass drying, only the ANN based model was used because ANFIS based model has a serious operational load in the training process. In energy and mass balance based model and ANFIS based model, the product quality is not used among the model outputs. Fig.7 shows the profiles of the product activity, which was obtained by drying model based on spatial distribution of quality and moisture inside the granule, according to the drying time and radial distance. The average product activity during drying is also presented in this figure with experimental data. As can be noted this figure, the product activity is retained at the surface of the granule due to the diffusion limitation of the drying process. Accordingly, the product activity is preserved in a thin layer at the surface of the granule. The product activity is decreased from the surface of the granule to the center. Fig. 8 shows that the comparison of real industrial data related to the change of the quality loss and simulation results of the ANN based quality model. For entire experimental data, it can be shown from this figure, the performance of the ANN based model is quite satisfactory.

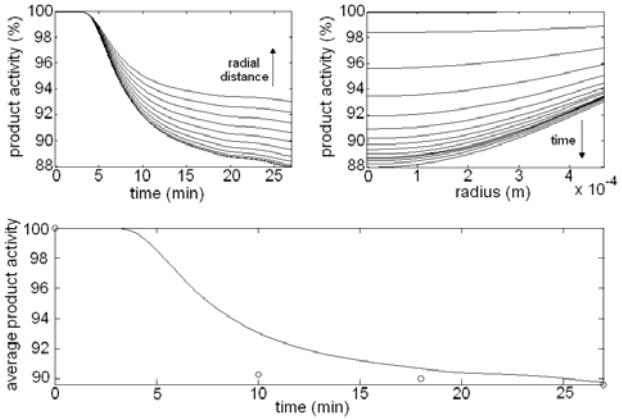


Fig. 7. The profiles of the product activity in the granule and average product activity for the model based on the spatial distribution of product activity. Experimental data (o), drying time 27 min, $R_0 = 5.10^{-4}$m, $X_0 = 1.563$ kg/kg, $T_0 = 16.9^oC$.

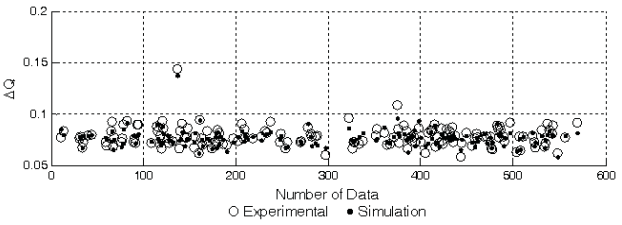


Fig.8. The change in the quality loss (ΔQ) obtained by ANN based quality model with experimental data.

The average root mean square error (RMSE) of ANN based quality model for five experimental data was calculated as 0.004742. For the second model based on spatial distributions of moisture and quality, the RMSE value is 0.005976. ANN based quality model has better performance than the

performance of the other model according to RMSE values. The coefficient of determination R^2 is the proportion of variability in a data set, as given below:

$$R^2 = \left(1 - \frac{SSE}{SST}\right) \tag{19}$$

SSE represents the sum of squared errors and *SST* denotes the total sum of squares. y_i and f_i denote a data set and the modeled values, respectively, and $\overline{y}$ represents the mean of the modeled values. Table 1 represents the R^2 values associated with all of the model approaches for only one experimental data selected randomly from database. An R^2 of 1.0 indicates that the regression line perfectly fits the data.

Table 1. The performance results of the all model approaches

Model	Dry matter of product R^2	Product temperature R^2
Model_1	0.97925	0.62711
Model_2	0.98743	0.72643
Model_3	0.98387	0.85256
Model_4	0.98525	0.81776

Model_1: Model based on the mass and energy balance,
Model_2: Model based on the spatial distribution in granule,
Model_3: Model based on ANN
Model_4: Model based on ANFIS

As can be noted from Table 1, R^2 values are fairly good for the dry matter of product, but R^2 values related to the product temperature are different from the each other. ANN and ANFIS based model approaches have better performances than the others.

5. CONCLUSIONS

In this study, four different modeling structures were considered for an industrial scale drying process. In all of the models, dry matter of product and the product temperature were used as model outputs. As comparing to the mass and energy balance based model, the second drying model provides significantly improved predictions by providing spatial distributions of moisture and quality inside the granule. Besides, the model accurately predicts the change of granule size during drying by the effect of the shrinkage of the granule. In the production plant, the product quality is measured with offline laboratory conditions as the amount of carbon dioxide produced upon introduction of the yeast into dough per unit time. The product quality is only predicted in ANN based model and the model based on spatial distributions of moisture and quality. It has been provided that the product activity can be observed online by these proposed model structures such as a soft sensor. In contrast with the physical approach based modeling, ANN or ANFIS based modeling approaches does not need prior theoretical knowledge. In order to overcome modeling difficulties, easily self-updating modeling structures can be designed to capture all of the system's operating conditions, as well as details that may have escaped observation. The investigated modeling structures may be an alternative to predict many parameters, such as the moisture content, dry matter of product, product temperature and the product quality or quality loss, in the biomass drying process industry.

REFERENCES

Azeem, M. F., Hanmandlu, M. and Ahmad, N. (2000). Generalization of adaptive neuro-fuzzy inference systems. *IEEE Trans. on Neural Networks*, *11*(6), 1332-1346.

Castellanos, J. A., Palancar, M. C. and Aragon, J. M. (2002). Designing and optimizing a neural network for the modeling of a fluidized-bed drying process. *Industrial and Engineering Chemistry Research*, *41*, 2262-2269.

Ciesielski, K. and Zbicinski I. (2001). Hybrid neural modeling of fluidized bed drying process. *Drying Technology, 19*(8), 1725-1738.

Coumans, W.J. (1987). Power law diffusion in drying processes. Ph.D. Thesis, Technical University Eindhoven, The Netherlands.

Jang, J. S. R. (1993). ANFIS: Adaptive-network-based fuzzy inference system. *IEEE Transactions on Systems Man and Cybernetics*, *23* (3), 665-685.

Kanarya, A. (2002). Mathematical modelling of fluidized bed drying process, *MSc Thesis*, Gebze Institute of Technology, Gebze, Turkey (in Turkish).

Köni, M., Türker, M., Yüzgeç, U., Dinçer, H. and Kapucu, H. (2009). Adaptive modeling of the drying of baker's yeast in a batch fluidized bed. *Control Engineering Practice*, *17* (4), 503-517.

Lievense, L.C. (1991). The inactivation of Lactobacillus Plantarum during drying. Ph.D. Thesis, Wageningen University, The Netherlands.

Liou, J.K., Luyben, K. Ch. A. M. and Bruin, S. (1984). A simplified calculation method applied to enzyme inactivation during drying. *Biotechn. Bioeng., 27*,109-116.

Quirijns, E.J., van Boxtel, A.J.B. and van Straten, G. (1998). The Use of moisture and temperature profiles in predicting product quality for optimal control of drying process. *Proceedings on Automatic Control of Food and Biological Processes IV*. Göteborg, Skjöldebrand, C. and G. Trystam (Eds.), Sweden, 485-490.

Quirijns, E.J., van Willigenburg, L.G., van Boxtel, A.J.B. and van Straten, G. (2000). The significance of modeling spatial distributions of quality in optimal control of drying processes. *Journal A. Benelux Quarterly Journal on Automatic Control*, *41*(3), 56-64.

Schoeber, W.J.A.H. (1976). Regular regimes in sorption processes.*Ph.D. Thesis*, Technical University Eindhoven, The Netherlands.

Temple, S. J., Tambala, S. T. and van Boxtel, A. J. B. (2000). Monitoring and control of fluid-bed drying of tea. *Control Engineering Practice*, *8*,165-173.

Temple, S.J. and van Boxtel, A.J.B. (1999). Modeling of Fluidized Bed Drying of Black Tea, *Journal of Agricultural Engineering Research*, *74,* 203-212.

Türker, M., Kanarya, A., Yüzgeç, U., Kapucu, H. and Şenalp Z. (2006). Drying of baker's yeast in batch fluidized bed. *Chemical Engineering and Proc.*, *45* (12), 1019-1028.

Yüzgeç, U., Türker, M. and Becerikli, Y. (2008). Modelling spatial distributions of moisture and quality during drying of granular baker's yeast. *The Canadian Journal of Chemical Engineering*, *86* (4), 725-738.

Dynamic Modeling and Control Issues on a Methanol Reforming Unit for Hydrogen Production and Use in a PEM Fuel Cell

Dimitris Ipsakis[1,2], Spyros Voutetakis[1], Panos Seferlis[3], Simira Papadopoulou[1,4]

[1]Chemical Process Engineering Research Institute (C.P.E.R.I.), CEntre for Research and Technology Hellas (CE.R.T.H.), P.O. Box 60361, 57001 Thermi-Thessaloniki, Greece (Tel: +30-2310-498 317; e-mail:paris@cperi.certh.gr).

[2]Department of Chemical Engineering, Aristotle University of Thessaloniki, P.O. Box 1517, 54124 Thessaloniki, Greece (Tel: +30-2310-498 353; e-mail:ipsakis@cperi.certh.gr).

[3]Department of Mechanical Engineering, Aristotle University of Thessaloniki, P.O. Box 484, 54124 Thessaloniki, Greece (Tel: +30-2310-994 229; e-mail:seferlis@cperi.certh.gr).

[4]Department of Automation, Alexander Technological Educational Institute of Thessaloniki, P.O. Box 14561, 54101 Thessaloniki, Greece (Tel: +30-2310-498 319; e-mail:shmira@teithe.gr).

Abstract: The presented research work focuses on the mathematical description and control analysis of an integrated power unit that uses hydrogen produced by methanol autothermal reforming. The unit consists of a reformer reactor where methanol, air and water are co-fed to produce a hydrogen rich stream through a series of reactions. The hydrogen main stream is fed to a preferential oxidation reactor (PROX) for the reduction of CO at levels below 50ppm with the use of air. In the end, the PROX outlet stream enters the anode of a PEM fuel cell where power production takes places to serve a load demand. The operation of the two reactors is described by a combination of partial differential equations (mass and energy balances) and non-linear equations (kinetic expressions of the reactions), while the power production in the fuel cell is based on the inlet hydrogen flow and on operational characteristics. A simple case sceanrio is employed when a step change on methanol flowrate is imposed. Main target is to identify and analyze the changes occuring in the main variables of concern (H_2, CO and temperature levels) that affect the overall system operation. Based on the results, an insight on the challenging control scheme will be applied in order to identify possible ways of setting up a reliable and robust control structure according to the developed mathematical model.

Keywords: methanol reforming, preferential oxidation, hydrogen, PEM fuel cell, dynamic modeling

1. INTRODUCTION

Hydrogen can be considered an energy carrier for the future and when derived from renewable energy sources can be totally non-polluting when used in fuel cells (Ipsakis D. et al., 2008). Fuel cells advantages include (Larminie J. and Dicks A., 2003) the low operation temperatures (~ 80^{o}C), the CO_2 tolerance by the electrolyte, the fast cold start and their few moving parts, which enhances their role as back-up units in vehicles. Nevertheless, the main disadvantage of fuel cells refers to the hydrogen supply. Natural gas, gasoline and higher hydrocarbons have been proposed for hydrogen production via steam reforming, but methanol carries the most advantages from all (Lindström B. and Petterson L.J., 2001). Methanol is a liquid that does not require special conditions of storage, while it is also free from high reforming temperatures and sulphur oxides that are met in methane and gasoline reforming. Moreover, methanol has a high H:C ratio and no C:C ratio and thus, prevents the soot formation (Lindström B. and Petterson L.J., 2001), while biomass resources can be used to produce methanol (bio-methanol). Production of hydrogen from methanol can be achieved in three ways: (i) steam reforming of methanol, (ii) partial oxidation of methanol and (iii) autothermal reforming of methanol (Lindström B. and Petterson L.J., 2001). Autothermal reforming has the asset of eliminating the disadvantages of steam reforming (endothermic process which requires a heating source) and partial oxidation (highly exothermic process which leads to the formation of hot spots in the catalyst) by properly selecting the reactants ratios in such a way, so that adiabatic conditions can be achieved. One of the drawbacks of hydrocarbons reforming however, is the production of CO at high levels that degrade the electrochemical performance of low temperature PEM fuel cells. Several processes used for the minimization of CO content at acceptable levels (less than 50ppm) have been discussed in the past, where among them preferential oxidation is considered to be the simplest and the least expensive method (Cipitì F. et al., 2007).

Most of the simulation and modeling studies on methanol reforming, focus on each subsystem as an individual part. Steam reformers have been modelled using axial distribution models (Suh J. S. et al., 2007), while PROX reactors have also been modelled using 2D models (Cipitì F. et al., 2007). In one of the very few papers that deal with an integrated system, Stamps A. T. and Gatzke E.P. (2006), developed and implemented a system level model of a vehicular reformer PEM fuel cell stack power system without the use however, of a hydrogen purification system, while the dynamic operation results were not provided explicitly. As can be seen, literature references lack in studies that deal with the detailed mathematical description of all the involved subsystems and on their overall dynamic interactions. The objective of this study is to analyze the dynamic behaviour of the two reactors based on a step change imposed in methanol flowrate. The reactors temperature and CO content will be continuously monitored, since they mostly affect the power production of the fuel cell. In the end, an analysis on the demanding task of the control of such an integrated power system will be applied, in order to identify all the necessary actions need to be taken for the development of a robust control scheme.

2. DESCRIPTION OF THE INTEGRATED POWER SYSTEM

Figure 1, shows the overall process flow diagram of the methanol autothermal reforming unit.

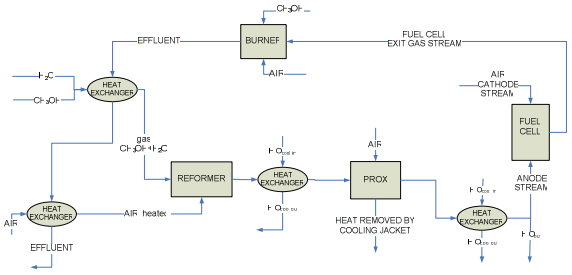


Fig. 1: Overall operation scheme of the methanol autothermal reforming unit.

Methanol, air and water are co-fed to the reformer (0.15m length and 0.1m diameter) after they are preheated by the effluent gases of a burner. The main reaction that takes place at the reformer is the steam reforming of methanol (R1), but because it is an endothermic reaction, oxygen also reacts with methanol (R2) to provide the necessary heat to the system.

Table 1. Reactions taking place at the reformer

$CH_3OH + H_2O \rightarrow CO_2 + 3H_2$ $\Delta H_{R,298}$ = 49kJ/mol, (R1)	$CH_3OH + 0.5O_2 \rightarrow CO_2 + 2H_2$ $\Delta H_{R,298}$ = -193kJ/mol, (R2)
$CO + H_2O \leftrightarrow CO_2 + H_2$ $\Delta H_{R,298}$ = -41.2kJ/mol, (R3)	$CH_3OH \rightarrow CO + 2H_2$ $\Delta H_{R,298}$ = 90.1kJ/mol, (R4)

The autothermal reforming of methanol is based on the effective selection of the reactants ratios at the inlet of the reformer. For the current study, it was found from steady-state simulations using Aspen Plus® (Ouzounidou M. et al., 2008a) that the optimum values for H_2O/CH_3OH and O_2/CH_3OH are between 1.5-1.7 and 0.1-0.15 respectively. Steam reforming is considered as the sum of the water gas shift (R3) and methanol decomposistion (R4) reactions.

The reduction of CO after steam reforming, takes place at the PROX reactor (0.1m length and 0.1m diameter). There, besides CO oxidation, H_2 oxidation also takes place, but it is kept at low rate through the effective selection of the catalyst. The reactions of the PROX can be seen in table 2:

Table 2. Reactions taking place at the PROX

$CO + 0.5O_2 \rightarrow CO_2$ $\Delta H_{R,298}$ = -283kJ/mol, (R5)	$H_2 + 0.5O_2 \rightarrow H_2O$ $\Delta H_{R,298}$ = -242kJ/mol (R6)

The main objective of the PROX reactor is to keep the CO concentration at a maximum limit of 50 ppm by using efficiently the O_2/CO ratio. From ASPEN Plus® simulations (Ouzounidou M. et al., 2008a), its optimum value is 1, but could vary severely with CO concentration. The PROX reactor due to the presence of the two highly exothermic reactions is surrounded by a jacket to cool the system from undesirable extreme heat generation. The kinetic expressions for all the involved reactions can be found in (Ouzounidou M. et al., 2008b) where an analysis on their selection is provided.

Finally, the hydrogen rich stream is fed to the anode of the fuel cell where it ionises releasing protons (H^+) and electrons (e^-). Protons pass through the proton exchange membrane to the cathode, while electrons flow through an external electric circuit and produce current. In the cathode, oxygen is fed via air supply and reacts with the protons and electrons to form water (Table 3).

Table 3. Reactions taking place at the PEM fuel cell

$H_2 \rightarrow 2H^+ + 2e^-$ Anode, (R6)	$0.5O_2 + 2H^+ + 2e^- \rightarrow H_2O$ Cathode, (R7)

3. MATHEMATICAL DESCRIPTION OF THE SUBSYSTEMS

In this section, the mathematical models of the involved subsystems will be presented and discussed. All parameters, variables and symbols are explained in the nomenclature section. gPROMS® was used to simulate the operation of the integrated power system (PSE®, 2002).

3.1 Reformer and PROX Reactors

The model equations for the catalytic reactors consist of the standard material and energy balances for a pseudo-homogeneous system (packed-bed type). These equations are in partial differential form with spatial and radial distributions. In order to simplify the model and reduce the computational effort needed to simulate all the subsystems, a number of assumptions have been made. It is highlighted, that the description of the system does not lose its accuracy

and the crucial phenomena can be easily described by the developed mathematical model.

- The ideal gas law is applied for all gas components.
- No diffusion phenomena are assumed to take place from the gas phase to the surface of the catalyst.
- Constant reactor pressure and fluid velocity.
- Constant physical properties (component density and heat capacity) over the range of conditions.
- The temperature in the cooling jacket of the PROX reactor is approximately uniform and the resistance to heat transfer occurs primarily between the reactor contents and the wall of the tube (being at the cooling medium temperature).

Material Balance Equation

$$\frac{\partial C_i}{\partial t} + u \cdot \frac{\partial C_i}{\partial z} - \varepsilon_{cat} \cdot D_z \frac{\partial^2 C_i}{\partial z^2} - \varepsilon_{cat} \cdot D_r \cdot \left(\frac{\partial^2 C_i}{\partial r^2} + \frac{1}{r} \cdot \frac{\partial C_i}{\partial r}\right) = \sum_{i=1}^{N} \sum_{j=1}^{R} \nu_{i,j} \cdot R_j \quad (1)$$

Energy Balance Equation

$$\sum_{i=1}^{N} \rho_i \cdot C_{p,i} \cdot \frac{\partial T}{\partial t} + \sum_{i=1}^{N} \rho_i \cdot C_{p,i} u \cdot \frac{\partial T}{\partial z} - k_z \frac{\partial^2 T}{\partial z^2} - k_r \left(\frac{\partial^2 T}{\partial r^2} + \frac{1}{r} \cdot \frac{\partial T}{\partial r}\right) = -\sum_{j=1}^{R} R_j \cdot (\Delta H_{R,T,j}) \quad (2)$$

Coolant Energy Balance Equation

$$\rho_c \cdot V_c \cdot Cp_c \cdot \frac{\partial T_c}{\partial t} = F_c \cdot Cp_c \cdot (T_{c,in} - T_c) + Q \quad (3)$$

$$Q = U \cdot A \int_0^{Reactor Length} (T - T_c) \cdot dz \quad (4)$$

Ideal Gas Law

$$P_i \cdot V = n_i \cdot R \cdot T \Rightarrow P_i = C_i \cdot R \cdot T \quad (5)$$

Species Flowrate

$$F_i = C_i \cdot Q_o \quad (6)$$

$$Q_o = u \cdot S = \frac{\sum F_{i,in} \cdot R \cdot T_{in}}{P_{reactor}} \quad (7)$$

Boundary Conditions (Eqs. 1-4)

z=0

$C_i = C_{i,in}$ *and* $T = T_{in}$, r ε [0,R] (8)

z=L

$$\frac{\partial C_i}{\partial z} = 0 \text{ and } \frac{\partial T}{\partial z} = 0 \text{ , } \quad r \varepsilon [0,R] \quad (9)$$

r=0

$$\frac{\partial C_i}{\partial r} = 0 \text{ and } \frac{\partial T}{\partial r} = 0 \text{ , } \quad z \varepsilon (0,L) \quad (10)$$

r=R

$$\frac{\partial C_i}{\partial r} = 0 \quad and \quad -k_r \cdot \frac{\partial T}{\partial r} = h_w \cdot (T - T_c) \text{ , } z\varepsilon(0,L) \quad (11)$$

In the solution of the reformer, h_w is 0 and for the PROX ≠0. For the discretization of the distributions the method of centered finite difference (2nd order) was used. The discretization of the axial and radial distribution was performed for 50 and 5 intervals, respectively. More intervals showed that the results are not affected, but the increase in the required computational effort leads to inefficient solution procedure.

3.2 PEM Fuel Cell

Larminie J. and Dicks A., (2003), presented an equation that relates the power production with the hydrogen flow and the operational characteristics of the fuel cell:

$$P_{fc} = F_{H2} \cdot n_e \cdot F \cdot \eta_F \cdot V_{cell} \quad (12)$$

The hydrogen flow predicted by the reactors mathematical model is used in the above equation to predict the power production as a function of time. V_{cell} is usually around 0.7V/cell (Larminie J. and Dicks A., 2003).

The same authors also concluded that if all the enthalpy of reaction of a hydrogen fuel cell was converted into electrical energy then the output voltage would be 1.48V (water in liquid form) or 1.25V (water in vapour form). Therefore, the difference between the actual cell voltage and this voltage represents the energy that is converted into heat instead. For the vapour case in our fuel cell, heat is calculated as:

$$Q_{fc} = P_{fc} \cdot \left(\frac{1.25}{V_{cell}} - 1\right) \quad (13)$$

3.3 Model validation

Based on the experimental results presented in (Ouzounidou M. et al., 2008b), validation with the above mathematical model (kinetic parameters and the axial and radial distributed paramaters were estimated) has been performed, where it has been found that the model accurately simulates the performance of both reactors. Figures 2 and 3 show the comparison between the experimental data and mathematical model for the two reactors, respectively and the deviation between simulated and experimental values is within the expected error (less than 5%). Only in low PROX temperatures a higher deviation is detected but considered negligible, since PROX operates in temperatures >150°C.

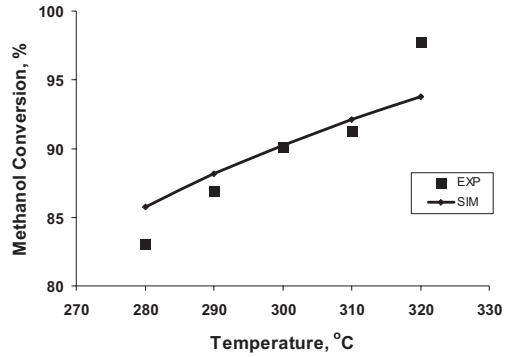


Fig. 2. Comparison between simulated and experimental results for the reformer

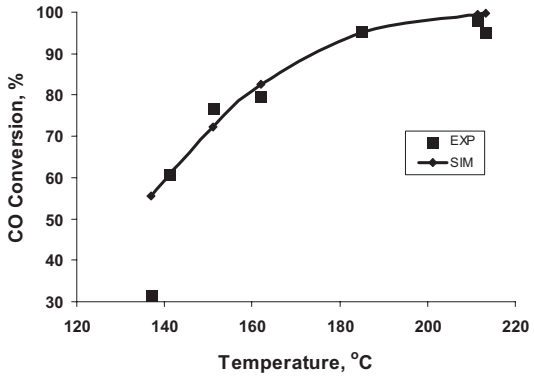


Fig. 3. Comparison between simulated and experimental results for the PROX

4. SIMULATION RESULTS FROM THE DYNAMIC OPERATION OF THE INTEGRATED POWER SYSTEM

The dynamic operation of the integrated power unit will be presented based on an imposed step change. The conditions of the simulated case study are: inlet methanol at 0.02mol/s, reformer temperature at 300°C, PROX temperature at 200°C, H_2O/CH_3OH at 1.5, O_2/CH_3OH at 0.14 and O_2/CO at 2. It is noted that the flowrates of the reactants and reactor temperature at the inlet of the reformer do not have a sharp constant value at the start of the simulation time, but a value that increases smoothly with time and reaches its steady state value after a few seconds. Fig.4 shows the 50% step change on the inlet methanol flowrate at t=30s (inlet water and oxygen flowrates are also increased based on the selected ratios). As can be seen, at the same time an increase at the exit hydrogen flowrate starts to appear and after 80s the increase in hydrogen flowrate is calculated at 49.6%, which shows that methanol conversion is practically unaffected by the increase in the methanol flowrate for the selected conditions.

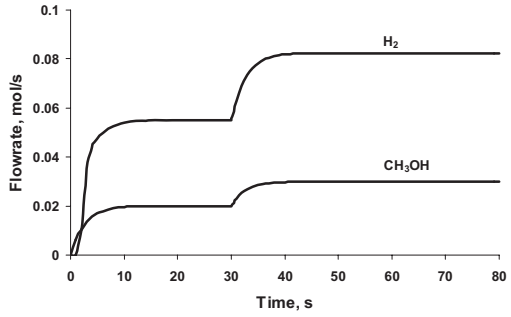


Fig. 4. Methanol inlet and hydrogen outlet flowrates

Similarly Fig.5 shows the CO content (ppm) where a 13% decrease is observed at the time that the step change occurs. This decrease is excpected due to the fact that water flowrate is increased (according to the methanol step change) and the water gas shift reaction is favored. This can also be concluded by the fact that CO flowrate increase is calculated at 22.5% which differs significantly from the 50% increase detected in other products, such as hydrogen.

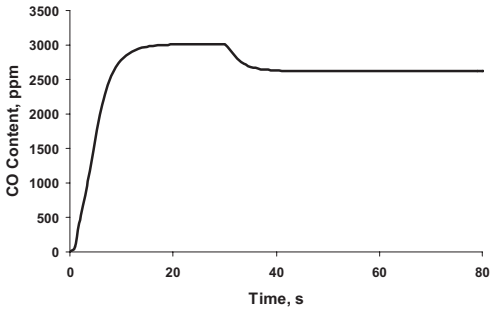


Fig. 5. Carbon monoxide content at the reformer exit

Fig.6 shows the reformer inlet and outlet temperature levels. The inlet gas temperature follows a smooth increase in order to prevent hot spots due to the rigorous exothermic partial oxidation of methanol.

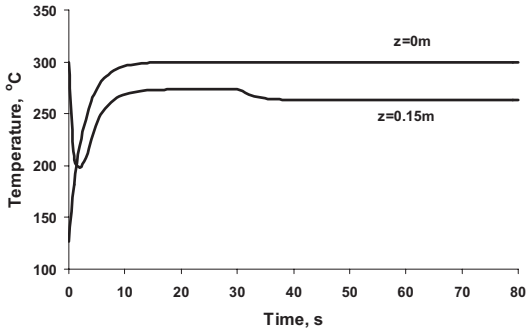


Fig. 6. Temperature levels at the reformer inlet and outlet

This smooth increase is based on the operation of the assumed preheater that heats the reactants mixture. On the

other hand, the outlet temperature of the reformer (at t=0s the reformer is assumed to be at 300°C) is initially decreased due to the fact that low temperature gas mixture exits the reactor at t<10s, but as the reformer operation proceeds, the exit temperature is gradually increased to 260-270 °C. As can be seen for the present conditions, the endothermic reactions prevail and the exit temperature is 30-40 °C lower than the inlet (at steady-state conditions) and also the step change seems to affect the temperature levels by lowering them by 10 °C.

Unlike the reformer, where negligible changes are observed in the radial domain, in the PROX the species concentration and temperature varies severely along the reactor radius. Fig.7 shows the CO content at the wall of the reactor, at its center and the average value that indicates the exit flow. As can be seen, the CO content is higher away from the center due to the lower temperature at that region (see Fig.8). As we move to the reactor center, the CO levels are quite low (practically zero at the reactor center) due to the increased temperature while the average value is always lower than 50ppm for the current conditions. It is highlighted that the CO at the exit of the reformer is measured and according to the selected O_2/CO ratio, the air feed rate is manipulated and introduced to the PROX reactor, so as to always provide a constant O_2/CO ratio. If the O_2 flow was constant based on the initial conditions, then the CO levels would have been higher indicating a possible fuel cell deterioration.

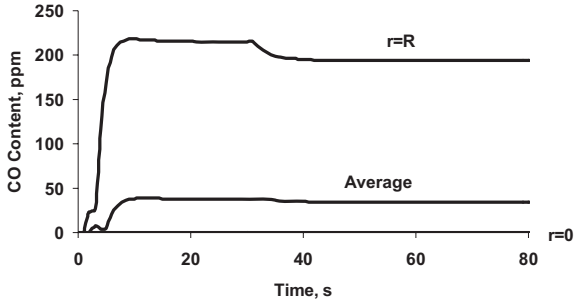


Fig. 7. Carbon monoxide content at the PROX wall, center and average value in the radial domain

Fig.8 shows the temperature levels at the wall of the reactor, at the center and the average (exit) value. Initially (t=0s), the reactor is assumed to be at 200°C and as the oxidations take place, the reactor center is found to have increased temperature in contrast with the reactor wall temperature that is maintained at low levels due to the presence of the cooling medium. Eventually, the exit (average) temperature is initially decreased, but as the gas mixture approaches the reactor exit, the temperature levels are increased. It can also been said, that at t=30s, a small increase is detected due to the increased flowrate at the inlet of the PROX which is not considered severe (less than 15°C). The presence of a controller to maintain the reaction temperature at specific levels is of primary importance and the manipulated value will be the coolant flowrate. It is noted that the hydogen main stream is assumed to be cooled down before entering the PROX and the fuel cell, but the dynamic simulation of the heat exchangers is omitted for this study, since we focus on the main subsystems.

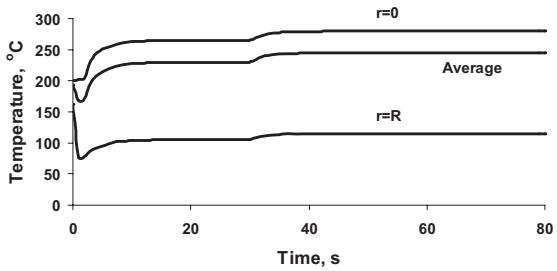


Fig. 8. Temperature levels at the PROX wall, center and average value in the radial domain

Finally, Fig. 9 shows the power and heat production levels based on the inlet hydrogen flow. As can be seen, at t=30s a 49.8% increase in power and heat production is detected, which should be taken into consideration at the developed control scheme that will always try to meet the load demand (see next section).

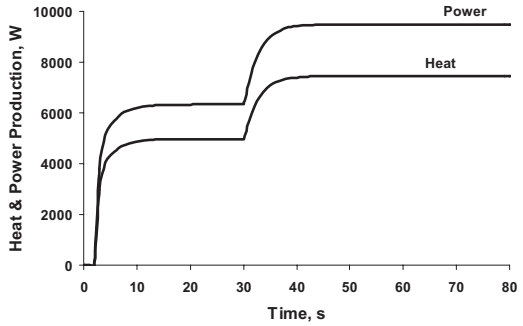


Fig. 9. Power and heat production in the fuel cell

5. CONTROL ISSUES ON THE INTEGRATED POWER SYSTEM

As it is obvious from the above analysis, the operation of the integrated power system, requires the development of a robust control scheme. The variables that constantly need to be monitored are: the reactor temperatures, the CO levels and finally, the power to be supplied to the load. In the reformer, the temperature will be controlled by the O_2/CH_3OH ratio and in the PROX through the coolant flowrate. The CO composition will be controlled by the effective selection of the H_2O/CH_3OH and O_2/CO ratios. Nevertheless, the main variable of concern is the power that needs to be provided to the load. Changes in the demanded load power level will be handled by manipulating the methanol flowrate in order to produce the hydrogen needed in the fuel cell to operate. As was presented, step changes in methanol flowrate affect the overall operation and model predictive control (MPC) is the more suitable control scheme of such an integrated power system. The control algorithm must also satisfy the bounds for CO and temperature levels in order to protect the various subsystems from deterioration (mainly PROX and fuel cell).

All these consequent changes will be decided based on the minimization of an objective function (MPC) that will take into account all the system necessary constraints. Special care, however, should be given to the fact that that slow and fast dynamics occur in the system (e.g. the fast PROX oxidation versus the slow coolant effect) that might need to be specially treated. Perturbation Theory is proposed to alleviate such problems, because it can be applied to mathematical systems that combine non-linear algebraic equations and differential ones (Kumar A. and Daoutidis P., 1999).

6. CONCLUSIONS

An integrated power system for the production of hydrogen via autothermal reforming of methanol has been studied in this paper. The developed mathematical model for the two reactors was validated and used for the simulation of the operation of the power unit where a step change was imposed. The next step will be the integration of the heat management system (burner and heat exchangers), while the developed control scheme will try to maintain the operational variables of concern at their desired values (set points).

Nomenclature

A:	heat transfer area, m^2
C_i:	concentration of the component i, mol/m^3
Cp_c:	coolant specific heat capacity, J/ K Kg
C_{pi}:	component i specific heat capacity, J/K kg
D_r:	radial effective diffusivity, m^2/s
D_z:	axial effective diffusivity, m^2/s
F:	Faraday's constant, Cb/mol
F_c:	coolant flowrate, kg/s
F_i:	flowrate of the component i, mol/m^3
h_w:	wall heat transfer coefficient, W/m^2 K
i:	component that takes part at the system
in:	inlet conditions
j:	number of reaction at the reactors
k_r:	radial thermal conductivity, W/m^2 K
k_z:	axial thermal conductivity, W/m^2 K
n_c:	number of cells of the PEM fuel cell
n_e:	number of electrons
n_F:	Faraday's efficiency, %
P_i:	parital pressure of the component i, bar
P_{fc}:	fuel cell power, Watt
$P_{reactor}$:	reactor pressure, bar
Q:	heat removed by the cooling jacket, Watt
Q_o:	volumetric flow, m^3/s
Q_{fc}:	heat, W
r:	radius of the reactor, m
R:	universal gas constant bar m^3/ mol K
R_j:	kinetic expression of the reaction j, mol/kg_{cat} s
S:	cross section of the reactor, m^2
t:	time, s
T:	temperature, K
T_c:	coolant temperature, K
u:	superficial gas velocity, m/s
U:	overall heat transfer coefficient, W/m^2 K
V_c:	coolant jacket volume, m^3
V_{cell}:	cell voltage, V/cell
z:	length of the reactor, m
$\Delta H_{R,T,j}$:	enthalpy of reaction j at temperature T, J/mol
ε_{cat}:	void fraction of the catalyst
$\nu_{i,j}$:	coefficient of the component i in the reaction j
ρ_i:	density of the component i, kg/m^3
ρ_c:	coolant density, kg/m^3

REFERENCES

Cipitì, F., Pino, L., Vita, A., Laganà, M., and Recupero V. (2007). Model-based investigation of a CO preferential oxidation reactor for polymer electrolyte fuel cell systems. *International Journal of Hydrogen Energy*, Volume 32, Pages 4040-4051

Ipsakis, D., Voutetakis, S., Seferlis, P., Stergiopoulos, F., Papadopoulou, S., and Elmasides C. (2008). The Effect of the Hysteresis Band on Power Management Strategies in a Stand-Alone Power System. *Energy*, Volume 33, Pages 1537-1550

Kumar A., and Daoutidis P. (1999). Control of nonlinear differential algebraic equation systems. Chapman & Hall/CRC, New York, Washington D.C.

Larminie J., and Dicks A. (2003). Fuel Cell Systems Explained. 2nd Edition, John Wiley & Sons Ltd

Lindström B., and Petterson L.J. (2001). Hydrogen generation by steam reforming of methanol over copper-based catalysts for fuel cell applications. *International Journal of Hydrogen Energy*, Volume 26, Pages 923-933.

Ouzounidou, M., Ipsakis, D., Voutetakis, S., Papadopoulou, S., and Seferlis P. (2008a). Experimental Studies and Optimal Design for a Small-Scale Autonomous Power System Based on Methanol Reforming and a PEM Fuel Cell. T*he AIChE 2008 Annual Meeting*, Philadelphia, PA, U.S.A., November 16-21, 2008

Ouzounidou, M., Ipsakis, D., Voutetakis, S., Papadopoulou, S., and Seferlis P. (2008b). A combined methanol autothermal steam reforming and PEM fuel cell pilot plant unit: Experimental and simulation studies, (submitted in Energy, Special Issue for PRES08)

Process Systems Enterprise Ltd, gPROMS Introductory User Guide, London, United Kingdom, Release 2.1.1.-15 March 2002.

Stamps, T., and Gatzke, E.P., (2006). Dynamic modeling of a methanol reformer—PEMFC stack system for analysis and design. *Journal of Power Sources*, Volume 161, Pages 356-370

Suh, J. S., Lee, M. T., Greif, R., and Grigoropoulos, C. P. (2007). A study of steam methanol reforming in a microreactor. *Journal of Power Sources*, Volume 173, Pages 458-466

Dynamic modelling of a three-phase catalytic slurry intensified chemical reactor

S. Bahroun*, C. Jallut*, C. Valentin*, F. De Panthou**

** Université de Lyon, F-69622, Lyon, France;*
Université Lyon 1, Villeurbanne;
LAGEP, UMR 5007, CNRS, CPE, 43, Bd du 11 Novembre 1918, 69100 Villeurbanne cedex, France;
(e-mail: bahroun@lagep.univ-lyon1.fr, Tel: (+33)4 72 43 18 81, Fax: (+33)4 72 43 18 99).
*** AETGROUP SAS, 6 montée du Coteau, 06800 Cagnes-sur-mer, France,*
(e-mail: fabrice.depanthou@aetgroup.com, Tel: (+33)6 24 05 25 83, Fax: (+33)4 92 08 53 40).

Abstract: Three-phases chemical reactions are widely applied industrially. They are highly non-linear, multivariable, exothermal processes. The aim of the present work is to propose a dynamic model of an intensified continuous three-phases mini-reactor. This model is developed to enable the transient phase monitoring of the mini-reactor. The reactor is treated as an association of J stirred tank reactors in series with back mixing effect. The model is used to describe the dynamic behaviour of the reactor during the hydrogenation of o-cresol on Ni/SiO_2 catalyst. The model consists in mass and energy balance equations for the catalyst particles, the gas and liquid bulk phases. The transient heat transfers between the metal body of the reactor, the coolant fluid and the bulk fluid are also taken into account.

Keywords: Intensification; Three-phase reactor; Hydrogenation; Dynamic modelling; Control; Dynamic behaviour.

1. INTRODUCTION

The pharmaceutical and the fine chemical industries produce molecules of high added values mainly in batch or fed-batch agitated reactors. Indeed, batch reactors, even if they provide the characteristics of flexibility and versatility required in the field of fine chemicals production, have a number of limitations: in particular, poor conditions of heat released by the chemical reactions leads to a serious safety problem for highly exothermal chemical reactions.

For some years, there is an alternative in the use of batch reactors, thanks to process intensification progresses and to mini-reactors development. The idea is to perform the reactions in continuous intensified reactors. On the one hand, the intensification leads to a better control of heat transfer that allows concentrating the reagent and thus limiting the quantities of solvent to be treated. On the other hand, intensification leads to reduce mass transfer limitations in the case of multiphase chemical reactors.

Nowadays, the control and safety of these reactors are important features in the design as well as in the operation of industrial processes. Such processes carry out complex reactions with constraints on thermal stability and/or selectivity as, for example, exothermal hydrogenation reactions (Vasco de Toldeo *et al.*, 2001).

Hydrogenation reactions are widely applied industrially. Development of efficient and reliable models for three phase reactors is still a difficult task because it involves many aspects including hydrodynamics, gas-liquid and liquid-solid mass transfer, heat transfer, reaction kinetics (Bergault *et al*, 1997).

In the literature, we can find many studies proposing dynamic modelling of catalytic three phase reactors (P.A Ramachandran and J.M. Smith, 1976; R.J. Wärnä and T. Salmi, 1995; Vasco de Toldeo *et al.*, 2001) but as far as we know, the problem of mini-reactors modelling is not addressed.

The aim of this paper is to propose a dynamic model of a continuous three-phases intensified mini-reactor. This model is developed to enable the transient period monitoring of the three phases mini-reactor. It will allow better predictions of the behaviour of the system and therefore secure and effective studies of the reactor control.

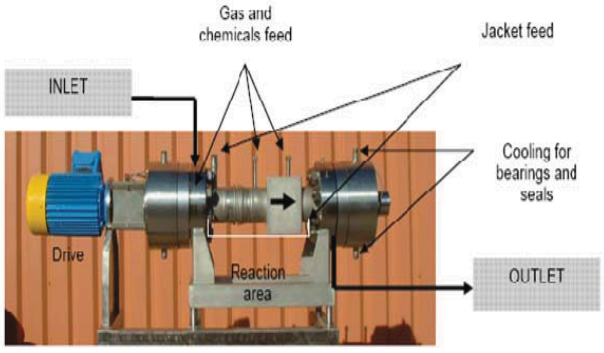


Fig.1. The "RAPTOR® (***R**éacteur **A**gité **P**olyvalent à **T**ransfert **O**ptimisé **R**ectiligne)*".

The mini-reactor under consideration, the "RAPTOR®" (presented by the figure 1) is developed by AETGOUP SAS Company a French society whose main activity is to offer a broad range of services mainly focused on chemical process industrialisation, in the field of pharmaceuticals, fragrances and aromas, cosmetics and specialty chemicals. For confidentiality reasons, we cannot give here a detailed description of the RAPTOR®.

2. MATHEMATICAL MODELLING

The model is used to describe the dynamic behaviour of the reactor during the hydrogenation of o-cresol on Ni/SiO_2 catalyst, taken as an example. A three-phase catalytic reactor is a system in which gas and liquid phases are in contact with a porous solid phase. The reaction occurs between a dissolved gas and a liquid-phase reactant in presence of a catalyst on the surface of the porous solid support (P.A. Ramachandran and R.V. Chaudhari, 1983), according to the following global stoichiometric equation:

$$A_{(g)} + \nu_B B_{(l)} \rightarrow \nu_C C_{(l)} \quad (1)$$

The chemical process involves several steps in series (see figure 2):

- diffusion of A from gas-liquid interface to the bulk liquid;
- diffusion of A and B from bulk liquid through liquid-solid interface;

- intraparticle diffusion of A and B into the pores of the solid;
- adsorption of the reactants;
- adsorbed phase reaction.

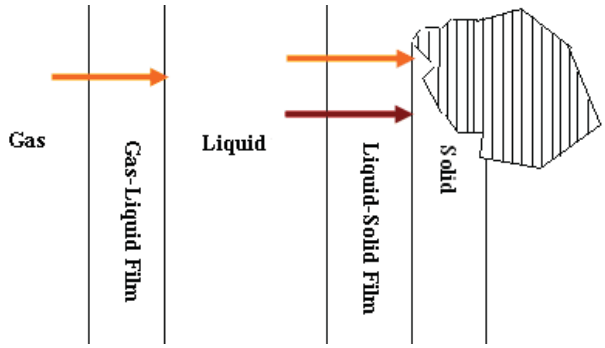


Fig.2. Steps of catalytic reaction.

The reactor is treated as an association of J stirred tank reactors in series with back mixing effect. Similarly, the jacket is treated as an association of J perfectly stirred tank reactors series. This way to consider the flows in the reactor is highly flexible and leads to a finite dimension model. Since an intensified reactor has to be compact, the influence of the reactor body itself may be significant with respect to the thermal transient behaviour of the system. Consequently, a piece of reactor body is associated to each perfectly mixed reactors used to model the flows (see figure 3).

The model consists in mass and energy balance equations for the catalyst particles, the gas phase and liquid-bulk phase, and also energy balance equations for the body of the reactor and for the coolant fluid into the jacket.

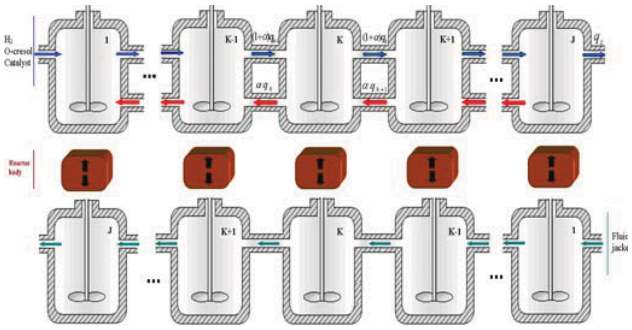


Fig.3. Flow model of the mini-reactor.

2.1 Kinetic model

Hydrogenation of o-cresol on Ni/SiO_2 catalyst (Hichri *et al.*, 1991) is taken as an example of chemical process that can be intensified. In this case, hydrogen reacts with o-cresol without any solvent due to the high heat and mass transfer capacities of the mini-reactor. The reaction can be represented by equation (1). In this case, A stands for hydrogen, B for o-cresol, C for 2-methylcyclohexanol.

Langmuir-Hinshelwood model represents in a realistic way the adsorption phenomena involved in heterogeneous catalysis processes. The reaction rate is calculated as follows (Hichri *et al*, 1991):

$$R = k\frac{K_A K_B C_{As} C_{Bs}}{(1+K_A C_{As})(1+K_B C_{Bs})} \quad (2)$$

C_{As} and C_{Bs} are respectively the hydrogen and o-cresol concentrations within the catalyst pores. The Arrhenius law gives the variation of the rate constants k with temperature while adsorption constants K_A and K_B variations derive from mass action law. The following expressions are taken from (Hichri *et al.*, 1991):

$$k(mol/kg.cat.s) = 5{,}46.10^{8}\exp(-82220/RT_s) \quad (3)$$

$$K_A(m^3/mol) = 10{,}55.10^{-3}\exp(+5003/RT_s) \quad (4)$$

$$K_B(m^3/mol) = 7{,}54.10^{-6}\exp(+16325/RT_s) \quad (5)$$

where $T_s\,(K)$ is the catalyst pellet temperature.

2.2 Mini-reactor model

The following assumptions are considered to derive the model (Vasco De Toledo *et al.*, 2001; Santana, 1999):

- the liquid and gas phases homogeneous suspension is considered as a pseudo-fluid with respect to the temperature;

- a global mass transfer coefficient is used to represent hydrogen transfer from the liquid surface to the bulk. Equilibrium conditions at the liquid surface are assumed;

- the pressure variations are negligible;

- the resistances to mass and heat transfer at the catalyst pellet surface and within the pores are lumped into global heat and mass transfer coefficients;

- the material balance in the gas phase, that is assumed to be pure hydrogen, is written at steady state.

Mass balance of reactant A in the gas phase

$$F_{Ag}^{0k} + V_k k_l a^k \left(C_{Al}^{k*} - C_{Al}^k\right) = F_{Ag}^{Ik} \tag{6}$$

Mass balance of reactant A in the liquid phase

$$\varepsilon_l^k V_k \frac{dC_{Al}^k}{dt} = (1+\alpha) q_l^{k-1} C_{Al}^{k-1} + \alpha q_l^{k+1} C_{Al}^{k+1} - (1+2\alpha) q_l^k C_{Al}^k - V_k k_s a\left(C_{Al}^k - C_{As}^k\right) + V_k k_l a^k \left(C_{Al}^{*k} - C_{As}^k\right) \tag{7}$$

Mass balance of reactant A in the solid phase

$$\varepsilon_s V_k \frac{dC_{As}^k}{dt} = (1+\alpha) q_s^{k-1} C_{As}^{k-1} + \alpha q_s^{k+1} C_{As}^{k+1} - (1+2\alpha) q_s^k C_{As}^k + V_k k_s a\left(C_{Al}^k - C_{As}^k\right) - \nu_A \varepsilon_s \rho_s V_k R(C_{As}^k, C_{Bs}^k, T_s^k) \tag{8}$$

Mass balance of reactant B in the liquid phase

$$\varepsilon_l^k V_k \frac{dC_{Bl}^k}{dt} = (1+\alpha) q_l^{k-1} C_{Bl}^{k-1} + \alpha q_l^{k+1} C_{Bl}^{k+1} - (1+2\alpha) q_l^k C_{Bl}^k - V_k k_s a (C_{Bl}^k - C_{Bs}^k). \tag{9}$$

Mass balance of reactant B in the solid phase

$$\varepsilon_s V_k \frac{dC_{Bs}^k}{dt} = (1+\alpha) q_s^{k-1} C_{Bs}^{k-1} + \alpha q_s^{k+1} C_{Bs}^{k+1} - (1+2\alpha) q_s^k C_{Bs}^k + V_k k_s a (C_{Bl}^k - C_{Bs}^k) - \nu_B \varepsilon_s \rho_s V_k R(C_{As}^k, C_{Bs}^k, T_s^k) \tag{10}$$

Energy balance in the fluid (gas + liquid) phase

$$V_k (\varepsilon_g^k \rho_g C_{pg} + \varepsilon_l^k \rho_l C_{pl}) \frac{dT_f^k}{dt} = hS (T_m^k - T_f^k) + V_k h_s a_{ls} (T_s^k - T_f^k) - (\varepsilon_g^k \rho_g C_{pg} + \varepsilon_l^k \rho_l C_{pl}) \left((1+\alpha) q_{k-1} (T_f^k - T_f^{k-1}) + \alpha q_{k+1} (T_f^k - T_f^{k+1})\right) \tag{11}$$

Energy balance in the solid phase

$$\varepsilon_s V_k \rho_s C_{ps} \frac{dT_s^k}{dt} = V_k h_s a_{l:s} (T_f^k - T_s^k) - (\rho_s C_{ps}) \left((1+\alpha) q_{sk-1} (T_s^k - T_s^{k-1}) + \alpha q_{sk+1} (T_s^k - T_s^{k+1})\right) - \varepsilon_s V_k \rho_s \Delta_r HR(C_{A,s}^k, C_{B,s}^k, T_s^k). \tag{12}$$

Energy balance of the body of the mini-reactor

$$(V_k \rho C_p)_m \frac{dT_m^k}{dt} = hS(T_f^k - T_m^k) + h_j S(T_j^k - T_m^k) \tag{13}$$

Energy balance of the refrigerant fluid into the jacket

$$(V_k \rho C_p)_j \frac{dT_j^k}{dt} = h_j S(T_m^k - T_j^k) - (\rho C_p)_j q_j^{k-1} (T_j^k - T_j^{k-1}). \tag{14}$$

3. SIMULATION RESULTS

Kinetic and thermodynamic parameters are taken from (Hichri *et al.*, 1991). The other physical parameters are taken from (Vasco de Toledo *et al.*, 2001; Adriano Pinto Mariano *et al.*, 2005).

We have performed a sensitivity study to define the optimum conditions that enable to achieve a high conversion rate under the constraint of the thermal runaway. One can see on the figure 5 the dynamic behaviour of the reactor and on the figure 4 the steady-state conversion profile that we have obtained. The operating conditions resulting from the sensitivity analysis lead to a conversion up to 90% at the reactor outlet, which is a very common industrial target.

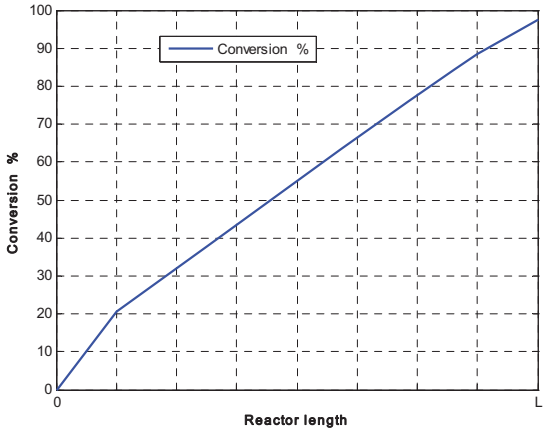


Fig.4. The steady-state conversion profile along the mini-reactor.

In figure 5, the dynamic behaviour of the reactor outlet temperature is represented. We observe that the fluid temperature is limited and increases by 24 %.

The catalyst temperature is slightly higher than that of reactant fluid since the reaction takes place in the pores catalyst. We also observe that the jacket temperature remains nearly constant while the temperature of the reactor body heats up by 15%. In fact the reactor body tends to store up the energy released by the reaction; it therefore appears that the reactor can work in hard reactive conditions, but safely.

When the optimum operating conditions have been established, we performed a steady-state characterization of the model; this feature allows not only steady analysis of model sensitivity with respect to the variation of input variables, but also consistency verification of the model.

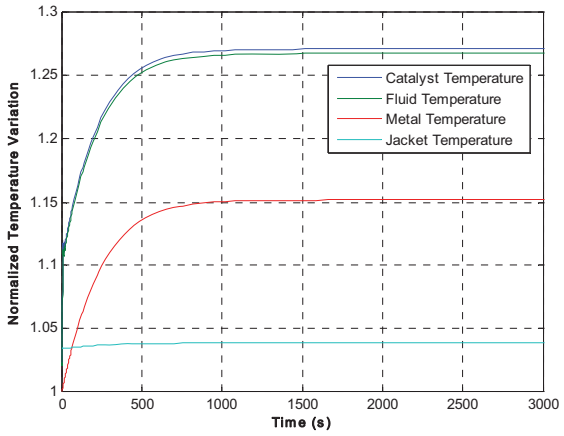


Fig.5. The dynamic behaviour of outlet reactor temperatures.

As far as intensified reactors are concerned, where the reaction takes place at very high temperature, the jacket plays a dual role in the evolution of the reaction. On the one hand, it is related to safety problems since the jacket fluid allows cooling the temperature of the bulk in order to avoid thermal runaway. On the other hand, it allows the indirect control of the outlet conversion by controlling the temperature of the bulk.

Figure 6 illustrates the steady-state characterization of outlet temperatures and conversion with respect to the inlet jacket fluid temperature (variations ranging from -5% to 5% around T_{j0}).

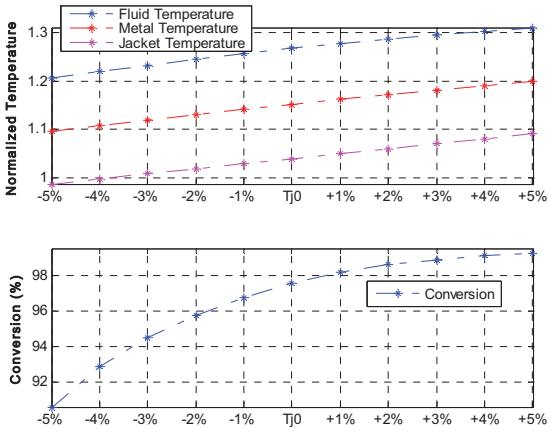


Fig.6. The steady-state characterization of outlet temperatures and conversion with respect to the inlet jacket fluid temperature.

From this figure, it is observed that both of outlet temperatures and conversion are very sensitive to changes in the inlet temperature of the jacket fluid. This sensibility of the dynamic behaviour of the reactor in relation to changes in the coolant fluid is observed mainly in industrial situations.

Figure 7 shows the steady characterization of outlet temperatures and conversion with respect to the inlet gas flow (ranging from -30% to 30% around q_{A0}). We can observe that the variations of q_{A0} have more impact on the conversion than the outlet temperatures.

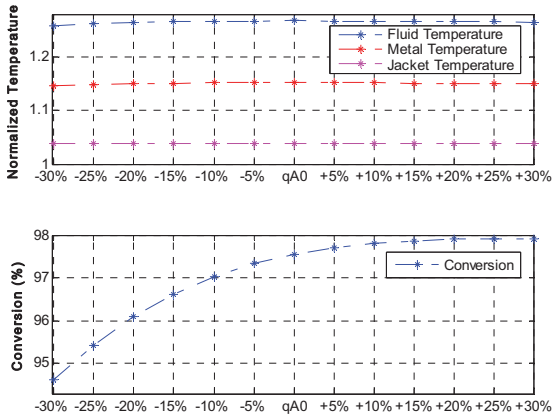


Fig.7. The steady-state characterization of outlet temperatures and conversion with respect to the inlet gas flow.

Figure 8 shows the effect of the inlet fluid temperature on the behaviour of outlet temperatures and conversion (variations ranging from -5% to 5% around $T_{f\,0}$). This figure points out that only conversion is sensitive to the variations of T_{f0}, as a result, disturbance on the inlet fluid temperature greatly affects the quality of the output product.

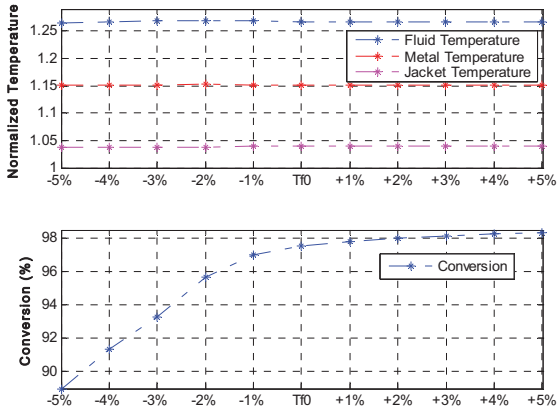


Fig.8. The steady-state characterization of outlet temperatures and conversion with respect to the inlet fluid temperature.

In figure 9, the steady-state characterization of outlet temperatures and conversion with respect to the inlet fluid flow (ranging from -30% to 30% around q_{f0}) is highlighted. We remark that the temperatures and conversion are very sensitive to q_{f0}. In fact, a decrease in the inlet fluid flow (-30%) allows obtaining a good conversion (=99.9%) with better thermal conditions. However this result leads to lower productivity.

According to the observations of these figures, the steady-state characterization of the model confirms the consistency of the model sets.

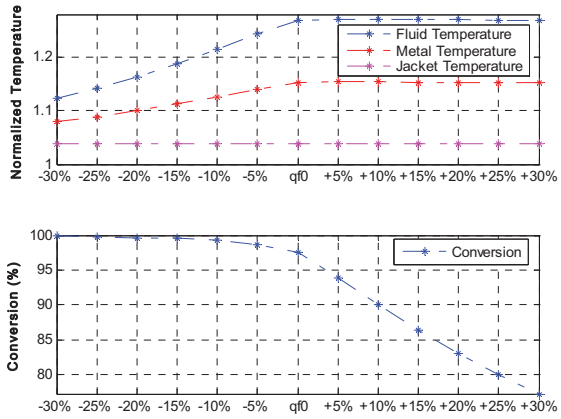


Fig.9. The steady-state characterization of outlet temperatures and conversion with respect to the inlet Fluid flow.

We observed that system's output variables are very sensitive to the variations of inlet jacket temperature. We could also highlight the fact that the conversion is very sensitive to the variations of inlet gas flow. Thus it may be concluded that these two variables may be chosen as an efficient control variables.

4. CONCLUSION

Reactions performed in multiphase catalytic intensified reactors have a complex behaviour due to heat and mass transfers and chemical kinetic interactions. The objective of the dynamic mathematical model developed in this work is to perform a detailed study of the process dynamic behaviour in order to define a suitable control structure. This structure has to control the outlet conversion and preserve safety conditions of the reactor. Others objectives such as high product quality and economy gain can also be considered.

ACKNOWLEDGMENTS

This project devoted to intensification of fine chemicals workshop is supported by the French research agency ANR (Agence Nationale de la Recherche). The authors thank warmly Shi Li for his discussions.

NOTATION

a	interfacial area, m^{-1}
C_A	concentration of the component A, mol m^{-3}
C_A^*	solubility of the component A, mol m^{-3}
C_B	concentration of the component B , mol m^{-3}
C_p	heat capacity, J K^{-1} kg^{-1}
F	molar flow, mol s^{-1}
h	heat transfer coefficient, W m^{-2} K^{-1}
k	kinetic constant, mol kg^{-1} s^{-1}
K	adsorption constant, m^3 mol^{-1}
k_l	mass transfer coefficient gas–liquid, m s^{-1}
k_s	mass transfer coefficient liquid–solid, m s^{-1}
q	volume flow, m^3 s^{-1}
R	reaction rate, mol kg^{-1} s^{-1}
S	surface, m^2
T	temperature, K
v	volume, m^3

GREEK LETTERS

$\Delta_r H$	heat of reaction, j mol^{-1}
ν	stoichiometric coefficient
α	back mixing
ρ	density, kg m^{-3}
ε_g	gas hold up
ε_l	bulk hold up
ε_s	solid hold up

SUBSCRIPT

A	component A
B	component B
j	jacket fluid
f	fluid
g	gas
k	reactor number
l	liquid
m	metal
s	solid

SUPERSCRIPT

k	reactor number
I	Inlet
O	Outlet

REFERENCES

Bergault, I., Rajashekharam, M. V., Chaudhari, R. V., Schweich, D. and Delmas, H. (1997). Modeling and comparison of acetophenone hydrogenation in trickle-bed and slurry airlift reactors. *Chemical Engineering Science*, 52, 4033–4043.

De Toledo, E. C. V., De Santana , P. L., Wolf Maciel, M. R., Filho , R. M. (2001). Dynamic modelling of a three-phase catalytic slurry reactor. *Chemical Engineering Science*, 56, 6055–6061.

Hichri, H., Armand, A., and Andrieu, J. (1991). Kinetics and slurry-type reactor modelling during catalytic hydrogenation of o-cresol on Ni/SiO2. *Chemical Engineering Process*, 30, 133–140.

Julcour, C., Stüber, F., Le Lann, J. M., Wilhelm, A. M., and Delmas, H. (1999). Dynamics of a three-phase up-flow fixed-bed catalytic reactor. *Chemical Engineering Science*, 54, 2391–2400.

Pinto Mariano A., De Toledo, E. C. V., Da Silva, J. M. F., Wolf Maciel, M. R. and Filho ,R. M.(2005). *Computers and Chemical Engineering*, 29, 1369–1378

Ramachandran, P. A., and Chaudhari, R. V. (1983). *Three-phase catalytic reactors*. Gordon and Breach, New York.

Ramachandran, P. A., and Smith, J. M. (1977). Dynamics of three phase slurry reactor. *Chemical Engineering Science*, 32, 873–880.

Santana, P. L., De Toledo, E. C. V., Meleiro, L. A. C., Scheffer, R., Freitas Jr., N. B., Wolf Maciel, M. R., and

Filho, R. M. (2001). A Hybrid mathematical model for a three-phase industrial hydrogenation reactor, *European symposium on computer aided process engineering*, Vol. (11), 279–284.

Wärnä, J., and Salmi, T. (1996). Dynamic modelling of catalytic three phase reactors. *Computers in Chemical Engineering*, 20(1), 39–47.

Identification of an Ill-Conditioned Distillation Column Process using Rotated Signals as Input

M.S. Sadabadi and J. Poshtan

Electrical Engineering Department, Iran University of Science and Technology, Tehran, Iran
(e-mails: sadabadi@iust.ac.ir, jposhtan@iust.ac.ir)

Abstract: The standard uncorrelated test signals estimate the low-gain direction of ill-conditioned multivariable systems poorly. Therefore, the low-gain information needs to be excited more. In this paper, identification of an ill-conditioned distillation column process using rotated signals is proposed. Rotated input signals allow more excitation to be applied in the weak gain direction of the process and less excitation in the strong gain direction. In this approach, the singular value decomposition (SVD) of the steady state gain matrix is used to rotate the input signals along the directions of the right singular vectors. Simulation results show good accuracy of the proposed method in identifying low and high gain directions.

Keywords: Rotated Input Design, Ill-conditioned Process, System Identification, Subspace Method, Distillation Column.

1. INTRODUCTION

Unlike SISO processes, MIMO processes may show ''directions'' (in the input vector space) in which the (steady-state or dynamic) effect of the inputs on the process outputs is much larger than in other directions (Zhu *et al.*, 2001, 2006). In such situations the process is said to be ill-conditioned. An ill-conditioned problem is a specific problem for multivariable processes.

Control-relevant identification of an ill-conditioned system requires special techniques. The directionality of such systems should be taken into account in the identification test signal design. Traditional uncorrelated open-loop step tests tend to excite the system mostly in high-gain directions. Therefore, the input test signals should be selected correctly (Zhu *et al.*, 2001, 2006).

In MIMO processes, the existing input test signal design methods can be divided in two categories, sequential input testing and simultaneous input testing (Conner *et al.*, 2004).

In sequential input testing, one signal, often PRBS (pseudo random binary sequence) or GBN (generalized binary noise) signal, is applied to each input separately while the other inputs are kept at their nominal values (Conner *et al.*, 2004). This input excitation usually takes a long time because the inputs are perturbed one at a time (Li *et al.*, 2008).

Simultaneously input testing excites more than one input at a time (Conner *et al.*, 2004). This method leads to more efficient use of the plant testing time (Conner *et al.*, 2004). Gevers *et al.* (Gevers *et al.*, 2006) using variance analysis shows that it is better to excite all inputs simultaneously. However, simultaneous uncorrelated open-loop tests cannot usually excite the ill-conditioned processes in low-gain direction (Zhu *et al.*, 2001, 2006). In these systems, the information of low gain direction is dominated by the noise (low SNR) and no good identification results can be achieved. This problem is caused by poor data not related to identification methods or model structure (Zhu *et al.*, 2001, 2006).

In order to increase the SNR in the low gain direction, one can replace the standard uncorrelated PRBS or GBN inputs with highly correlated signals as inputs. Koung and MacGregor (Koung *et al.*, 1993) proposed rotated inputs. These signals allow more excitation to be applied in the weak gain direction of the process and less excitation in the strong gain direction (Conner *et al.*, 2004). In their approach, the singular value decomposition (SVD) of the steady state gain matrix is used to rotate the input signals along the directions of the right singular vectors (Li *et al.*, 2008). Therefore, for constructing a rotated input signal, preliminary knowledge of the steady-state gain matrix is needed (Conner *et al.*, 2004).

In this paper, MIMO rotated input design for an ill-conditioned distillation column process identification is proposed.

The paper is organized as follows: In Section 2 and 3, ill-conditioned processes and rotated input design are respectively described. In Section 4, the application of the proposed method is carried out on high-purity distillation column as an ill-conditioned process. Finally, section 5 concludes the paper.

2. ILL-CONDITIONED PROCESS

Consider the multivariable (MIMO) system with n inputs and n outputs as follows

$$y(j\omega) = G(j\omega)u(j\omega) \tag{1}$$

The singular value decomposition (SVD) of G can be written (Skogestad *et al.*, 2001) as:

$$G(j\omega) = U(j\omega)\Sigma(j\omega)V^H(j\omega) \tag{2}$$

where U and V are the left and the right singular unitary matrices, respectively. Matrix Σ is the singular value matrix which is diagonal containing the singular values σ_i in decreasing order. The complex frequency $j\omega$ denotes that the SVD in general is a frequency dependent measure. For 2×2 processes, we can write (Jacobsen, 1994)

$$U = [\bar{u}\ \ \underline{u}],\quad \Sigma = diag(\bar{\sigma},\underline{\sigma}),\quad V = [\bar{v}\ \ \underline{v}] \tag{3}$$

and

$$G\bar{v} = \bar{\sigma}\bar{u},\ G\underline{v} = \underline{\sigma}\,\underline{u} \tag{4}$$

where $\bar{\sigma}$ denotes the maximum gain of G (in terms of 2-norm), $\bar{v}$ and $\bar{u}$ are the corresponding input and output directions, respectively. Similarly, $\underline{\sigma}$ is the minimum gain of G with corresponding input direction $\underline{v}$ and output direction $\underline{u}$. Note that the singular values and the corresponding input and output directions are frequency dependent; however, for simplicity $j\omega$ is omitted.

The condition number of gain matrix G is given by the ratio of the upper and lower singular values as follows (Jacobsen, 1994)

$$\gamma(G) = \bar{\sigma}/\underline{\sigma} \tag{5}$$

A process is said to be ill-conditioned if $\gamma(G) >> 1$ in some frequency range (Jacobsen, 1994).

In these processes, the process gain is strongly dependent on the direction of the input vector (Jacobsen, 1994). Therefore, the response of the plant is much stronger if input vector is in the high gain direction than if it lies along the low gain direction (Jacobsen, 1994). This can cause difficulties in the identification of ill-conditioned processes. In other words, ill-conditioned processes represent one of the most difficult kinds of linear processes to be identified (Micchi *et al*. 2008).

The ill-conditioned processes also have strongly interactions. The relative gain array (RGA), proposed by Bristol (Bristol, 1966), is a valuable criterion for evaluating the degree of interactions or directionality. The elements of the RGA is defined as follows (Zhu *et al.*, 2006)

$$\lambda_{ij}(j\omega) = g_{ij}(j\omega)[G^{-1}(j\omega)]_{ji} \tag{6}$$

where g_{ij} is the i,j element of G. As the elements in each row and column in the RGA adds up to unity, it is sufficient to consider the 1,1 element for the 2×2 case (Jacobsen, 1994). When one refer to the RGA, it means the 1,1 element of the RGA, i.e., λ_{11}. Large value of λ_{11} denotes that the process is strongly interactive (Jacobsen, 1994).

Note that there are differences between strongly interactive and ill-conditioned processes. A strongly interactive process is always ill-conditioned while the opposite is not always true (Jacobsen, 1994).

3. ROTATED INPUT DESIGN

Consider the singular value decomposition (SVD) of the steady state gain matrix $\bar{G}$.

$$\bar{G} = \bar{U}\,\bar{\Sigma}\,\bar{V}^H \tag{7}$$

Using the above equation, the steady state output of process, with N input-output data, can be written as (Conner *et al.*, 2004)

$$\bar{Y}^T = \bar{G}\hat{U}^T = \bar{U}\,\bar{\Sigma}\,\bar{V}^H\hat{U}^T \tag{8}$$

where

$$\bar{Y} = [\bar{Y}_1\ \bar{Y}_2 \dots \bar{Y}_n] \tag{9a}$$

$$\hat{U} = [U_1\ U_2 \dots U_n] \tag{9b}$$

and

$$\bar{Y}_i = [\bar{y}_i(1)\ \bar{y}_i(2) \dots \bar{y}_i(N)]^T \tag{10a}$$

$$U_j = [u_j(1)\ u_j(2) \dots u_j(N)]^T \tag{10b}$$

In theses equations, an overbar indicates steady state of a variable if inputs are held constant from the current time forward.

Now, the original inputs, $\{U_i\}$, are scaled by the singular values $\{\sigma_i\}$ to give new inputs $\tilde{U}$ (Conner *et al.*, 2004).

$$\tilde{U} = [U_1\ \ U_2(\frac{\sigma_1}{\sigma_2})\ \ U_3(\frac{\sigma_1}{\sigma_3})\ \ \dots\ \ U_n(\frac{\sigma_1}{\sigma_n})] \tag{11}$$

Then the rotated inputs are produced as follows (Conner *et al.*, 2004)

$$\Xi = \alpha\tilde{U}V^H \tag{12}$$

where α is a factor that should be adjusted so that the outputs do not exceed prespecified limits.

By using the rotated inputs, Ξ, instead of the original inputs, $\hat{U}$, equation (8) can be rewritten as (Conner *et al.*, 2004)

$$\bar{Y}^T = \alpha\bar{U}\,\bar{\Sigma}\,\tilde{U}^T \tag{13}$$

Therefore, the modes of the steady-state gain matrix are individually excited by scaled, uncorrelated PRBS or GBN signals (Conner *et al.*, 2004).

A generalization of the rotated inputs design procedure to non-square multivariable systems of arbitrary dimensions is presented at Micchi *et al*., 2008.

4. CASE STUDY: HIGH-PURITY DISTILLATION COLUMN

A binary distillation column as in Fig. 1 is considered as an ill-conditioned process. The column is running in LV-configuration.

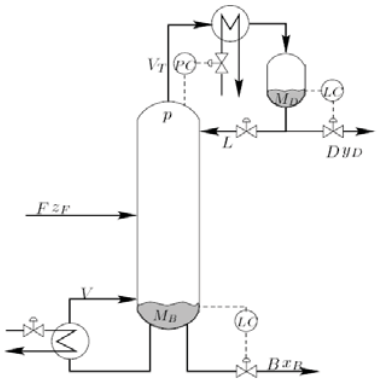


Fig. 1. High-purity distillation column

In this study, reflex (L) and boilup (V) flow rates are considered as the inputs and distillate (y_d) and bottom (x_b) compositions are considered as the outputs of the distillation column. For more detailed description, one can refer to Skogestad, 1997.

High-purity distillation is a challenging process application for system identification because of its nonlinear and strongly interactive dynamics (Rivera *et al*., 2007). Despite their nonlinear behavior, the ability to control high-purity distillation columns using linear controllers is desirable in practice for reasons of simplicity (Rivera *et al*., 2007). Thus, in many studies like this study, a linearized model is used.

The linear model of a distillation column can be described as (Jacobsen, 1994)

$$\begin{bmatrix} y_1(j\omega) \\ y_2(j\omega) \end{bmatrix} = G(j\omega) \begin{bmatrix} u_1(j\omega) \\ u_2(j\omega) \end{bmatrix} \tag{14}$$

where the state space model of *G* is as follows

$$\dot{x} = \begin{bmatrix} -0.0051 & 0 & 0 & 0 & 0 \\ 0 & -0.0737 & 0 & 0 & 0 \\ 0 & 0 & -0.1829 & 0 & 0 \\ 0 & 0 & 0 & -0.4620 & 0.9895 \\ 0 & 0 & 0 & -0.9895 & -0.4620 \end{bmatrix} x + \begin{bmatrix} -0.629 & 0.624 \\ 0.055 & -0.172 \\ 0.030 & -0.108 \\ -0.186 & -0.139 \\ -1.230 & -0.056 \end{bmatrix} u \tag{15a}$$

$$y = \begin{bmatrix} -0.7223 & -0.5170 & 0.3386 & -0.0163 & 0.1121 \\ -0.8913 & 0.4728 & 0.9876 & 0.8425 & 0.2186 \end{bmatrix} x \tag{15b}$$

Fig.2 and Fig.3 respectively show the singular values and RGA plotted as functions of frequency for the high-purity distillation column.

It can be seen that the process has large condition number and high RGA in the low frequency range.

At steady state, the high-gain singular value and the low-gain singular value are equal to $\bar{\sigma} = 198.2$ and $\underline{\sigma} = 1.36$, respectively. Steady state condition number is $\gamma \approx 146$.

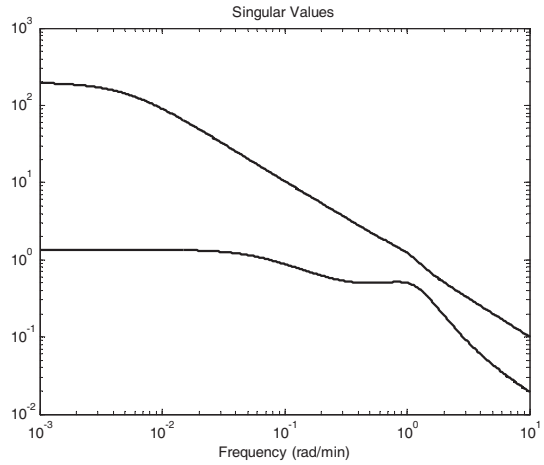


Fig. 2. Singular values of the distillation column

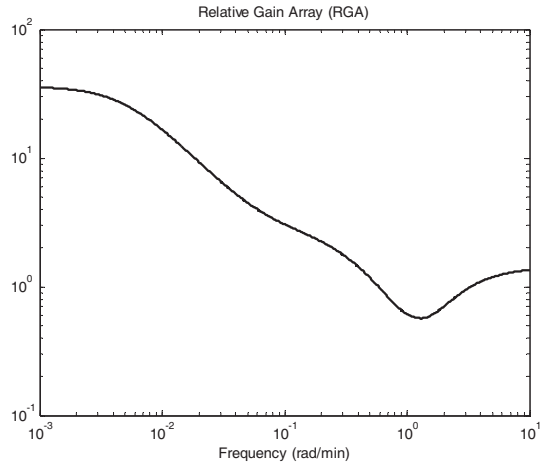


Fig. 3. RGA of the distillation column

Therefore, the largest effect on the outputs is obtained by moving the inputs in opposite directions which causes the two outputs to move in the same direction. The smallest effect is obtained by moving the inputs in the same direction which moves the two outputs in opposite directions.

4.1 Input Test Signals

For identification of the high purity distillation column, two open-loop test signals are considered: uncorrelated signals and rotated signals as outlined in Section 3.

Identification data are constructed by using generalized binary noise (GBN) signals as inputs, both uncorrelated and correlated in the case of rotated inputs. GBN signals, proposed by Tulleken (Tulleken, 1990), have many favorable features, in particular in terms of frequency content, which is typically superior to that of pseudo-random binary noise (PRBS) and of step signals (Zhu, 2001).

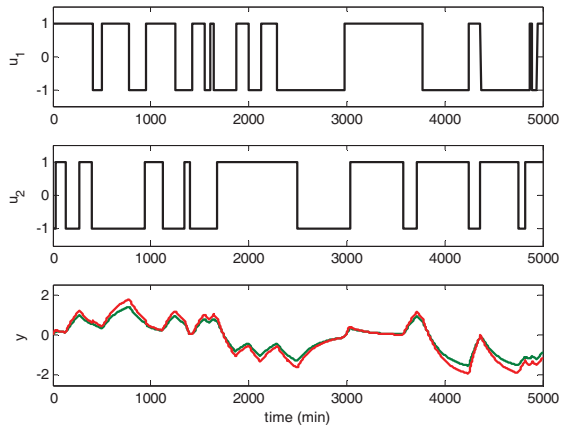


Fig. 4. Data collected using uncorrelated GBN signals as inputs

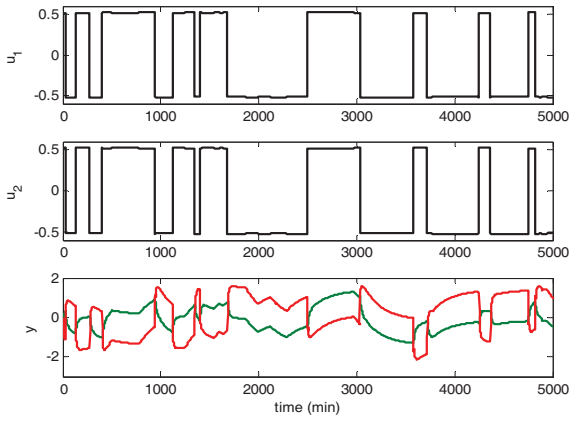


Fig. 5. Data collected using rotated GBN signals as inputs

Hence, two independent GBN signals with amplitude ± 1, switching time $T_{sw} = 300\,\text{min}$, and final time $T_f = 5000\,\text{min}$ are applied on the inputs of the plant simultaneously. The sampling time is selected to be 1 min.

Normally, distributed output noise with a signal-to-noise ratio (SNR) of 10 is added to both outputs. The input-output data from the two experiments are shown in Fig. 4 and Fig. 5.

Fig. 6 and Fig. 7 show the excitations of output directions in the uncorrelated and the correlated tests.

It can be seen that the uncorrelated test inputs only excite the high gain direction. In other words, in this case, the outputs of the process have no information about the low-gain direction of the model. It is clear that the low-gain direction information needs to be excited more in order to obtain good estimates of model. Therefore, strongly correlated test inputs with larger amplitudes are needed (see Fig. 7).

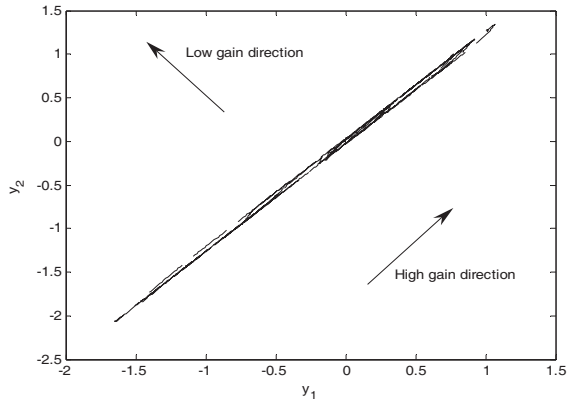


Fig. 6. Excitation of output directions in the uncorrelated test

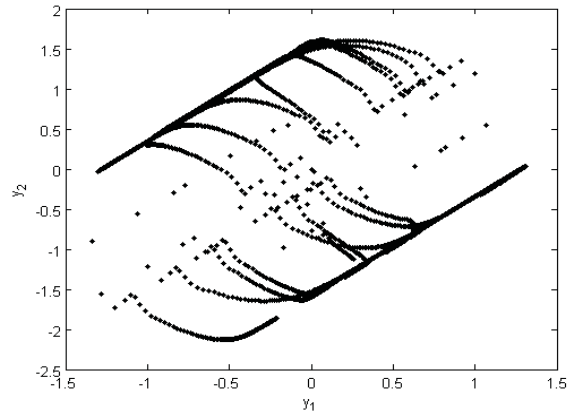


Fig. 7. Excitation of output directions in the rotated test

4.2 Subspace Identification

In this paper, a MIMO structure for the model is considered. In other words, there is a common model for all outputs. System identification is performed using subspace identification (SID) method. This method involves particular matrices obtained from output and input data and performs projection operations to cancel out the noise contributions. Thus, the system model is obtained in state-space form using these projected data matrices. A detailed treatment of this method can be found in Van Overschee *et al.*, 1996.

For identifying the system using subspace method, model order should be selected. In this paper, the model order is determined using number of nonzero singular values of matrix M given by (Misra *et al.*, 2003)

$$M = Y_f \,/_{U_f} W_p \prod\nolimits_{U_f^T}^{\perp} \tag{16}$$

$$= Y_f \prod\nolimits_{U_f^T}^{\perp} \left[[W_p \prod\nolimits_{U_f^T}^{\perp}]^{+} [W_p \prod\nolimits_{U_f^T}^{\perp}] \right]$$

where

$$Y_f \underset{U_f}{/} W_p = [Y_f \prod\nolimits_{U_f^T}^{\perp}][W_p \prod\nolimits_{U_f^T}^{\perp}]^{+} W_p;\ W_p = \begin{bmatrix} U_p \\ Y_p \end{bmatrix} \tag{17a}$$

$$\prod\nolimits_{U_f^T}^{\perp} = I - \prod\nolimits_{U_f^T}^{\perp} = I - U_f (U_f^T U_f)^{+} U_f^T \tag{17b}$$

(Y_f, U_f) and (Y_p, U_p) are future and past output-input data, respectively, that is:

$$Y_f = \begin{bmatrix} y_r & y_{r+1} & \cdots & y_{r+M-1} \end{bmatrix} \tag{18a}$$
$$U_f = \begin{bmatrix} u_r & u_{r+1} & \cdots & u_{r+M-1} \end{bmatrix}$$

$$Y_p = \begin{bmatrix} y_0 & y_1 & \cdots & y_{M-1} \end{bmatrix} \tag{18b}$$
$$U_f = \begin{bmatrix} u_0 & u_1 & \cdots & u_{M-1} \end{bmatrix}$$

where r is greater than the system order n ($r > n$) and $M = N - 2r + 1$. The order of system is determined using singular value decomposition of matrix M in equation (16) as follows:

$$M = \begin{bmatrix} \hat{Q}_s & \hat{Q}_n \end{bmatrix} \begin{bmatrix} \hat{S}_s & 0 \\ 0 & \hat{S}_n \end{bmatrix} \begin{bmatrix} \hat{V}_s^T \\ \hat{V}_n^T \end{bmatrix} \tag{19}$$

In the absence of noise, the rank of this matrix is exactly *n*. Thus there are exactly *n* nonzero singular values in the SVD in equation (19). In the presence of noise, however, the data matrix on the left hand side of (19) becomes a full rank matrix. The selection of the sizes of $\hat{S}_s$ and $\hat{S}_n$ then requires determining which singular values can be considered small, hence essentially zero, and which ones large. If the noise level is not too high, there is usually a significant difference between noise and signal singular values, and their separation is easily achieved (Misra *et al*., 2003). However, for ill-conditioned systems, even a small magnitude of noise can make it very difficult to determine the system order correctly (Misra *et al*., 2003).

For solving this problem, additional requirements must be posed on the input signals. In other words, input signals must excite ill-conditioned system in order to produce output signals as uncorrelated as possible. Therefore, rotated input signal is one of the most appropriate input signals to be used in subspace identification of ill-conditioned multivariable systems.

Fig.8 shows the singular values of the matrix M (for $r = 6$) in the rotated inputs case. As can be observed in this figure, the system order is 5.

After model order determination, subspace method MOESP is used to identify the system. Identification for both types of input test is performed over 20 simulation runs. For estimated model validation, the singular values and RGA are checked. Fig. 9-12 show the singular values and RGA of 20 simulation runs.

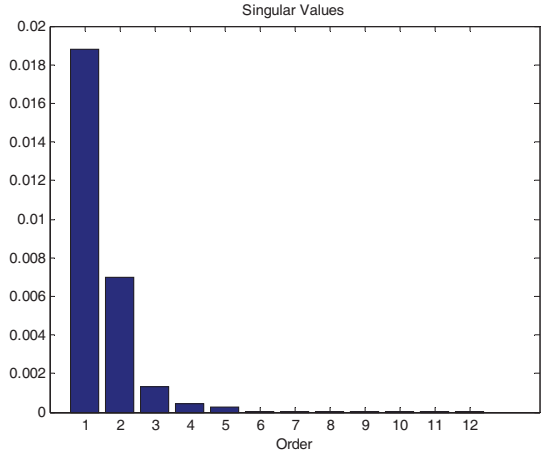


Fig. 8. Singular-value plot for distillation column using rotated input test

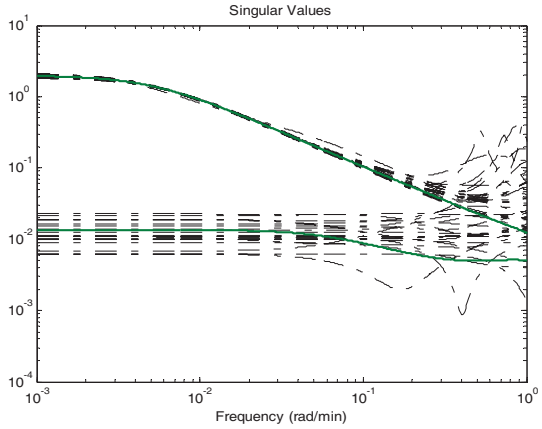


Fig. 9. Singular values of MIMO MOESP models from 20 simulations using uncorrelated test

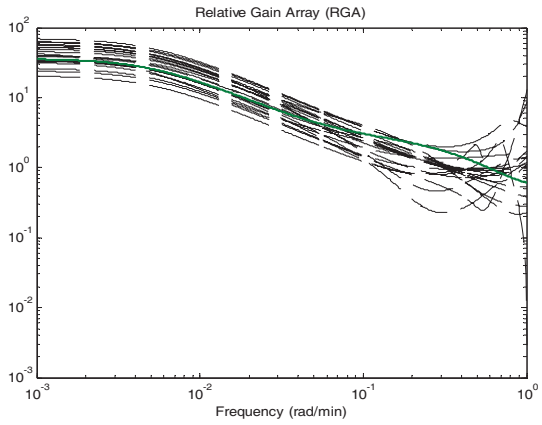


Fig. 10. RGA of MIMO MOESP models from 20 simulations using uncorrelated test

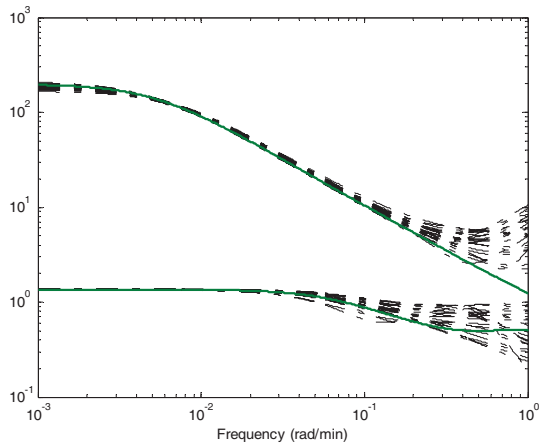


Fig. 11. Singular values of MIMO MOESP models from 20 simulations using rotated input test

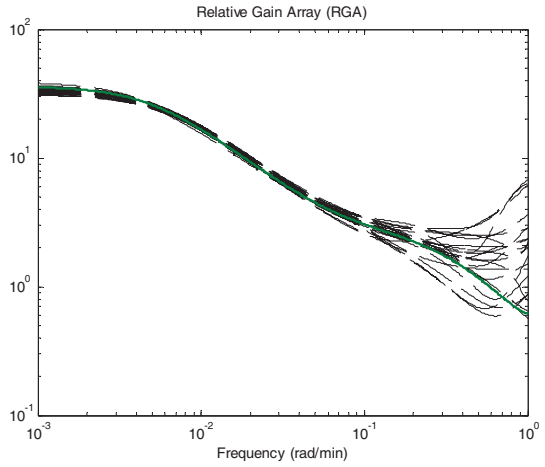


Fig. 12. RGA of MIMO MOESP models from 20 simulations using rotated input test

Note that in these figures, solid lines are the true values and dashed lines are the estimates. It can be seen that the high gain of process is easily estimated whereas the low-gain is very poorly estimated using the uncorrelated test signals. However, by using rotated input signal, good estimates of both low and high gain directions are achieved.

5. CONCLUSIONS

The standard uncorrelated test signals estimate the low-gain direction of ill-conditioned systems poorly. Therefore, the low gain information needs to be excited more in order to obtain good estimates. In this paper, MIMO rotated input design for ill-conditioned process identification was described. Rotated input signals allow more excitation to be applied in the weak gain direction of the process and less excitation in the strong gain direction. In this approach, the singular value decomposition (SVD) of the steady state gain matrix is used to rotate the input signals along the directions of the right singular vectors. The application of the proposed method was carried out on a high-purity distillation column as an ill-conditioned process. Simulation results show good accuracy of the proposed method in identifying both low and high gain directions.

REFERENCES

Bristol, E.H. (1966). On a new measure of interactions for multivariable process control. *IEEE Trans. Automatic Control*, AC-11, 133-134.

Conner, J.S. and Seborg, D.E. (2004). An evaluation of MIMO input design for process identification. *Ind. Eng. Chem. Res.*, 43, 3847-3854.

Gevers, M., Miskovic, L., Bonvin, D., and Karimi, A. (2006). Identification of multi-input systems: variance analysis and input design issues. *Automatica*, 42, 559-572.

Jacobsen, E.W. (1994). Identification for control of strongly interactive plants. *AIChE Annual Meeting*, San Francisco.

Koung, C.W. and MacGregor, J.F. (1993). Design of identification experiments for robust control. A geometric approach for bivariate processes. *Ind. Eng. Chem. Res.*, 32, 1658-1666.

Li, T. and Georgakis, C. (2008). Dynamic input signal design for the identification on constrained systems. *Journal of Process Control*, 18, 332-346.

Micchi, A. and Pannocchia, G. (2008). Comparison of input signals in subspace identification of multivariable ill-conditioned systems. *Journal of Process Control*, 18, 582-593.

Misra, P. and Nikolaou, M. (2003). Input design for model order determination in subspace identification. *AIChE Journal*, 49, 2124-2132.

Rivera, D.E., Lee, H., Mittelmann, H.D., and Braun, M.W. (2007). High-purity distillation: using plant-friendly multisine signals to identify a strongly interactive process. *IEEE Control Systems Magazine*, 28, 72-89.

Skogestad, S. and Postlethwaite, I. (2001). *Multivariable feedback control, Analysis and design*, Second Edition, John Wiley & Sons.

Skogestad, S. (1997). Dynamic and control of distillation columns- A tutorial introduction. *Trans. IChemE*, 75, 539-562.

Tulleken, H.J.A.F. (1990). Generalized binary noise test-signal concept for improved identification-experiment design. *Automatica*, 26, 1, 37-49.

Van Overschee, P. and Demoor, B. (1996). *Subspace identification for linear systems: theory-implementation-applications*, Kluwer, Dordrecht, Netherlands.

Zhu, Y. and Stec, P. (2006). Simple control-relevant identification test methods for a class of ill-conditioned processes. *Journal of Process Control*, 16, 1113-1120.

Zhu, Y. (2001). *Multivariable system identification for process control*, Elsevier Science.

A Sampling based method for linear parameter estimation from correlated noisy measurements

Ugur Guner*. Jay H. Lee**
Matthew J. Realff***

**Georgia Institute of Technology, Atlanta, GA 30332 USA (Tel: 404-388-2149; e-mail: Ugur.Guner@chbe.gatech.edu).*
***Georgia Institute of Technology, Atlanta, GA 30332 USA (e-mail: Jay.Lee@chbe.gatech.edu)*
**** Georgia Institute of Technology, Atlanta, GA USA (e-mail: matthew.realff@chbe.gatech.edu)*

Abstract: We address the problem of linear parameter estimation in discrete time state space models in the presence of serially correlated error in variables. The common way to solve parameter estimation problem is least squares (LS) methods. LS method is not considered to be effective when both dependent and independent variables are contaminated by noise. Total Least Squares (TLS) has been introduced as the method for parameter estimation in the case of noisy response and predictor variables. However, TLS solution is not optimal when number of data is limited and noise is correlated. Constrained TLS is a variant of TLS that considers correlation of noise in the data as additional constraints. We introduced a novel method based on a stochastic sampling method to solve estimation problem from correlated noisy measurements, and we compared it with the existing methods through in silico examples. Our method demonstrates significant improvement over other common estimation algorithms, LS, TLS and Constrained TLS under the different amount of correlated noise and data points. It has the potential to be the valuable tool for the difficult real life problems, such as, biological systems where data is limited and noisy.

Keywords: Linear parameter estimation; Least Squares; Total Least Squares; Constrained Total Least Squares; Multiplicative noise; additive noise; correlation; state space models; stochastic.

1. INTRODUCTION

The problem of linear parameter estimation arises in a broad class of scientific disciplines such as signal processing, automatic control, system theory, general engineering, statistics, physics, economics, biology, medicine, etc (Huffel ,1991). Linear estimation problem becomes challenging in the presence of correlated noise. Errors are unavoidable and can be related to many sources, such as modelling, human or instruments. They may appear in different forms depending on the source of error and nature of the system. Noise can be proportional to the signal itself (multiplicative),simply additive or it can include both components. In this paper, we will particularly focus on parameter estimation in linear discrete time state space models in the presence of measurements that include both multiplicative and additive error terms. One can write the linear discrete time system as follows;

$$\hat{x}_i^{k+1} = a_{i1}\hat{x}_i^k + a_{i2}\hat{x}_2^k +,...,+a_{iN}\hat{x}_N^k \qquad i = 1,...,N \tag{1}$$

In this equation this equation, value of observed state at time $k+1$ is linear function of all N observed states at time point, k . This model can be extended for all states and time points in a compact form as follows:

$$\hat{X}' = A^{(N\times N)}\hat{X} \tag{2}$$

$$\hat{X}'^{(N\times(M-1))} = \left\{\vec{x}^{\,2},..., \vec{x}^{\,M}\right\} \qquad \hat{X}^{(N\times(M-1))} = \left\{\vec{x}^{\,1},..., \vec{x}^{\,(M-1)}\right\}$$

where M is the number of time points and N is the number of states. Each column of X and X' is represented with the vector,

$$\vec{x}^{\,j} = \left[\hat{x}_1^{\,j},..., \hat{x}_N^{\,j}\right]^T . \tag{3}$$

The parameters are collected in matrix,

$$A^{(N\times N)} = \left\{ a_{ij} \right\}. \tag{4}$$

This paper is organized as follows. In next section we will briefly summarize common methods to solve linear estimation problem. Furthermore, we will introduce our novel approach based on a sampling algorithm. Section 3 will summarize assessment of the performance of our method compared to some common existing methods. Finally, section 4 will present the concluding remarks.

2. METHODS

2.1 Common methods

Many methods have been introduced to solve the linear estimation problem (Ljung ,1987 and Huffel ,1991). The classic way to solve the linear estimation problem is least squares. In the classical least squares regression theory, the errors are assumed to be confined only to $\hat{X}'$ (response variables) , and $\hat{x}$ (predictor variables) are assumed to be error free. One can write least squares estimate for parameters as follows,

$$A^T = \left(\hat{X}\hat{X}^T \right)^{-1} \hat{X}\hat{X}'^T \tag{5}$$

However, in our problem, it is not realistic to assume $\hat{X}$ to be error free as it shares the same columns with $\hat{x}'$except for the first column. (See (2)). This results in serial correlation between $\hat{X}$, and $\hat{x}'$.

Total least squares (TLS) is another method of linear parameter estimation when there are errors in both sides of the equation ($\hat{X}$ and $\hat{x}'$) (Huffel ,1991).

$$\hat{X}' = X' - \Delta X' \qquad \hat{X} = X - \Delta X$$

$$\Delta X' = \left[\Delta x^2,...,\Delta x^M\right] \qquad \Delta X = \left[\Delta x^1,...,\Delta x^{M-1}\right] \tag{6}$$

where, $\Delta X'$ and ΔX are the noise terms. Since X and X' are not known, for each state ($i = 1,...,N$), equation (2) can be written in the following format;

$$x_i + \Delta x_i = a^{(1\times N)}\left(\hat{X} + \Delta\hat{X}\right)$$

(7)

where $x_i = \left[x_i^2,...,x_i^M\right]$ and $\Delta x_i = \left[\Delta x_i^2,...,\Delta x_i^M\right]$ are the i^{th} rows of $\hat{X}'$ and $\Delta X'$ respectively (See (2, 6)). $a = \left[a_{i1},..,a_{iN}\right]$ is the i^{th} row of A .

Let,

$$C = \left[\hat{X}^T \; x_i\right] \quad \text{,and} \quad \Delta C = \left[\Delta\hat{X}^T \; \Delta x_i^T\right] \tag{8}$$

Then, equation (6) can be written as;

$$\left(C + \Delta C\right)\cdot\begin{bmatrix} a^T \\ -1 \end{bmatrix} = 0$$

where a is the i^{th} row of A . The TLS problem then can be posed as follows;

$$\min\left\| \Delta C \right\|_2^F \quad \text{subject to} \quad (C+\Delta C)\cdot\begin{bmatrix} a^T \\ -1 \end{bmatrix} = 0 \tag{9}$$

The solution to this problem given as;

$$a^T = \left(\hat{X}\cdot\hat{X}^T - \lambda^2 I\right)^{-1}\hat{X}\cdot\hat{X}'^T \tag{10}$$

where λ is the smallest singular value of C . Compared to least squares solution (5), the TLS solution has a correction term, λ at the inverse of the matrix. This reduces the bias in the solution which is caused by noise in X (Kim et al, 2007). However, TLS solution inherently assumes that the noise terms, $\Delta X'$ and Δx_i are independent , which is not the case here.

The correlation between two noise term requires the total least squares solution to have additional constraints instead merely satisfying the existence of a solution (Cadzow and Wilkes, 1985). Recently, Kim et al. (2007) applied constrained least squares algorithm in the context of gene network identification problem on a linear discrete time model. In their model, they rewrite error term, ΔC in an open form and as follows;

$$\Delta C^{(M-1)\times(N+1)} = \left[\Delta\hat{X}^T \; \Delta x_i^T\right] = \begin{bmatrix} \Delta x_1^1 & ... & \Delta x_i^1 & .... & \Delta x_N^1 & \Delta x_i^2 \\ . & . & . & . & . & . \\ . & . & . & . & . & . \\ . & . & . & . & . & . \\ \Delta x_1^{M-1} & ... & \Delta x_i^{M-1} & ... & \Delta x_N^{M-1} & \Delta x_i^M \end{bmatrix} \tag{11}$$

They introduced the vector, $e_i^{(N\times 1)} = [0,...,1,..,0]^T$ whose elements are zero except for the i^{th} element, which is equal to 1. All error terms are rewritten in a vector form as follows;

$$\Delta Y^{1\times(N(M))} = \left[\left(\Delta x^1\right)^T,....,\left(\Delta x^M\right)^T\right] \tag{12}$$

The first N columns of ΔC can be written as follows;

$$\Delta C^i = G_i\cdot\Delta Y^T \tag{13}$$

Where $G_i = \left[\left(I_{M-1}\otimes e_i\right)^T \; 0_{(M-1)\times N}\right]$ and I_{M-1} denotes the identity matrix of size $(M-1)\times(M-1)$ and the symbol, $\otimes$, denote Kronecker product of two matrices. $0_{(M-1)\times N}$ represents the matrix of zeros with size, $(M-1)\times(N)$

After several steps and simplifications, they posed this as optimization problem as follows;

$$\min\left\|\Delta C\right\|^2 = \min_a\left[a \;\; -1\right] C^T\left(H_a^{-1}\right)^T\left(H_a^{-1}\right)C\begin{bmatrix} a^T \\ -1 \end{bmatrix} \tag{14}$$

where $H_a = \sum_i a_i\cdot G_i$. This is a nonlinear, non-convex optimization problem without constraints. They initialized the optimization problem with least squares solution.

2.2 Our method

In this paper, we introduced a new method to solve linear parameter estimation problem when both sides of the equation are contaminated by error. This method based on a stochastic sampling approach. In this method, observations $(\hat{X}, \hat{X}')$ are perturbed by adding a negative noise term in each sample.

$$ {}^{(j)}y_i^k = \hat{x}_i^k - {}^{(j)}\varepsilon_i^k \qquad {}^{(j)}\varepsilon_i^k \propto N\left(0, \sigma_i^k\right) \tag{15} $$

$$ i = 1,...,N \qquad j = 1,...,S \qquad k = 1,...,M $$

Where ${}^{(j)}y_i^k$ is the j^{th} perturbed observation for i^{th} state at time point k and ${}^{(j)}\varepsilon_i^k$ is the amount of perturbation sampled from Gaussian distribution of zero mean and σ_i^k variance in the j^{th} sample. N is the number of states, S is the number of samples and M stands for the number of time points. Variance for perturbation σ_i^k is chosen roughly close to variance of the observation error in i^{th} state and k^{th} time point. (μ_{ik}, see equation (20)). The amount of perturbation , ${}^{(j)}\varepsilon_i^k$, is selected through Monte Carlo Sampling procedure.

Finally, all data points for each state i at each sample j are collected in a vector form. This is performed for both sides of equation (2). Let,

$$ {}^{(j)}y_i = \left[{}^{(j)}y_i^1,...,{}^{(j)}y_i^k,...,{}^{(j)}y_i^{M-1}\right]^T $$

$$ {}^{(j)}y_i' = \left[{}^{(j)}y_i^2,...,{}^{(j)}y_i^k,...,{}^{(j)}y_i^M\right]^T \tag{16} $$

Equation (16) can be written in a matrix form for all states, $i = 1,...,N$

$$ Y^{(j)} = \left[{}^{(j)}y_1 \ldots {}^{(j)}y_i \ldots {}^{(j)}y_N\right]^T $$

$$ Y'^{(j)} = \left[{}^{(j)}y_1' \ldots {}^{(j)}y_i' \ldots {}^{(j)}y_N'\right]^T \tag{17} $$

Next, equation (17) can be written for all samples, $j = 1,...,S$;

$$ Y^{Total} = \left[Y^{(1)} \ldots Y^{(j)} \ldots Y^{(S)}\right] $$

$$ Y'^{Total} = \left[Y'^{(1)} \ldots Y'^{(j)} \ldots Y'^{(S)}\right] \tag{18} $$

The parameters are estimated using least squares solution;

$$ A^T = \left(\left(Y^{Total}\right)^T \cdot Y^{Total}\right)^{-1}\left(Y^{Total}\right)^T Y'^{Total} \tag{19} $$

3. ASSESSING THE PERFORMANCE

To test the performance of our method against least squares (LS), Total Least Squares , and Constrained TLS solutions, we created ensemble of 50 linear time discrete systems with different parameters, each consisting of 10 states. This is achieved by creating random $A^{10\times10}$ matrices. We assumed sparse structure for each $A^{10\times10}$ matrix, therefore the number of non-zero elements are fixed to 30 out of 100 total connections. Each system is simulated for certain number of time points (M) starting from a random initial condition. However, we assumed limited number of data for the systems, as most of the real systems have small number of data (Most biological systems, gene networks, etc.). Multiplicative and additive noise terms are added to the simulation results as follows;

$$ \hat{x}_{ik} = x_{ik} + x_{ik}\mu_{ik} + \eta_{ik} $$

$$ \mu_{ik} \approx N(0, \sigma_1) \qquad \eta_{ik} \approx N(0, \sigma_2) \tag{20} $$

In this equation, $\hat{x}_{ik}$ is the observed value of i^{th} state at k^{th} time point. ε_{ik} and η_{ik} are random variables assumed to have Gaussian distribution with zero mean and variances σ_1 and σ_2 respectively. The term, $x_{ik}\mu_{ik}$ corresponds to the multiplicative noise term, whereas η_{ik} stands for the additive noise term. Our algorithm is tested against the other algorithms for different level of noise and number of time points. Performances of methods for parameter estimation are quantified as the Frobenius norm of the deviation of estimated parameters from their true values relative to the Frobenius norm of true parameters according the following formula;

$$ E_A = \frac{\left\|A - A^R\right\|_F}{\left\|A^R\right\|_F} \tag{21} $$

where A^R and A stand for the true and estimated parameters respectively. In addition, fitness of the system is evaluated and compared to true values of states similar to (15);

$$ E_X = \frac{\left\|X - X^R\right\|_F}{\left\|X^R\right\|_F} \tag{22} $$

where X and X^R indicates estimated and true values of states, respectively. E_A and E_X are calculated for ensemble of 50 different systems at each number of sample point and averaged. In figure 1, one can see the comparison of the methods with respect to number of samples when multiplicative and additive noise terms are set to $\sigma_1 = 0.10$, and $\sigma_2 = 0.0001$ and number of time points is 10.

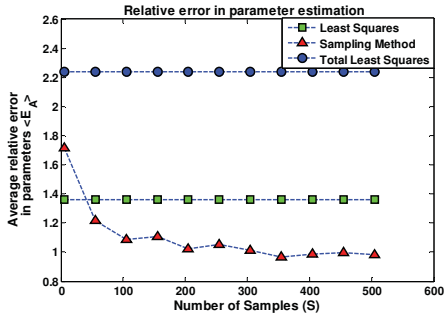


Fig.1. Relative error in parameter estimation vs number of samples for 10 time points at $\sigma_1 = 0.10$

Our method shows significant decrease in relative error in parameters compared to total least squares and least squares with increasing number of samples.

Methods	M=12	M=18	M=24
Least Squares	2730.1	1.1925	0.59776
Sampling Method **(at S=500)**	**0.11563**	**0.48125**	**0.44043**
Total Least squares	200.95	29.465	35.352
Constrained TLS	0.72001	289.98	3.7678

Table1. Average relative error in fitness for different methods for different time points at $\sigma_1 = 0.10$

Table 1 depicts the average relative error in fitness across the different methods and for different number of data. Our method outperforms all methods for 500 samples.

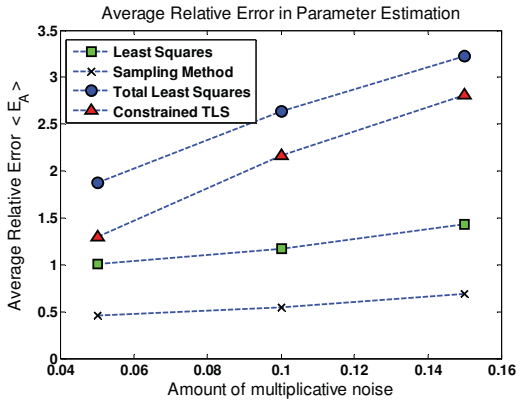


Fig.2. Average relative error in parameters versus amount of multiplicative noise for 18 time points

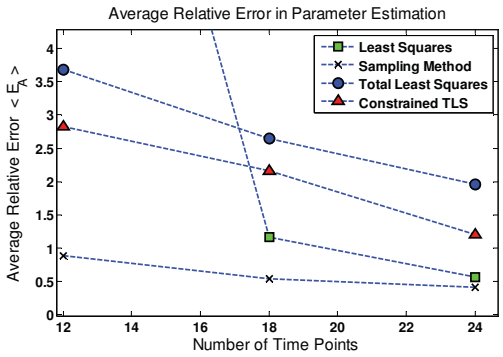


Fig.3. Average relative error in parameters versus number of time points at $\sigma_1 = 0.10$

Figure 2 indicates that our method performs significantly better than LS, TLS and constrained TLS across different levels of noise. TLS is expected to perform better in the case of noisy dependent and independent variables if enough data points are available. However, in this particular problem, due to the serially correlated multiplicative error and limited number of data, its performance is even below least squares solution. We assumed relatively small number of data, because for most of the interesting real systems, data is usually limited and noisy. Constrained TLS solution resolves the serial correlation problem and performs better than TLS, however, its performance still falls behind least squares solution because of limited data. When the number of data increases, the performances of all methods converge (See Figure (3)).

In Fig.3, one can observe that sampling method gives least amount of error in parameters at different number of time steps.

Methods	$\sigma_1 = 0.05$ $\sigma_2 = 0.0001$	$\sigma_1 = 0.10$ $\sigma_2 = 0.0001$	$\sigma_1 = 0.15$ $\sigma_2 = 0.0001$
Least Squares	7.3e+007	2730.1	9.9e+10
Sampling Method **(at S=500)**	**0.059484**	**0.11563**	**0.16913**
Total Least squares	11.331	200.95	21.864
Constrained TLS	1.2951	0.72001	8.4658

Table 2. Average relative error in fitness for different methods for 10 time points at different levels of noises.

In table 2, it is seen that the fitness of our method is much better than other methods for different noise levels.

4. CONCLUSIONS

The contribution of this work can be summarized in two ways. First, our method outperforms the common estimation methods in parameter estimation in the presence of correlated noise. Second, fitness of the estimated parameters through our method is significantly better than LS, TLS and Constrained TLS method. This method is particularly promising in the application of gene network identification problem. The biological measurements are notorious for having a high level of multiplicative noise which makes the network identification problem difficult. Our method has the potential to be the valuable tool for this difficult problem.

REFERENCES

Cadzow, J. A., Wilkes, D. M. (1985). *IEE international conference on Acoustics, Speech and Signal Processing,* 10, 1341-1344.

Huffel, S. V. (1991). *The total least squares problem: computational aspects and analysis,* Society for Industrial and Applied Mathematics, Philadelphia.

Kim, J., Bates, D. G., Postlethwaite, I., Harrison, P., and Cho, K. (2007). *BMC Bionformatics,* 8, 8.

Ljung, L. (1987). *System identification: theory for the user: computational aspects and analysis,* Prentice Hall information and system science series, Englewood Cliff, NJ.

Experimental and Modeling Studies for a Reactive Batch Distillation Column

Almıla Bahar*. Canan Özgen**

Department of Chemical Engineering, Middle East Technical University,
Ankara, 06531, Turkey
e-mail: *abahar@metu.edu.tr, **cozgen@metu.edu.tr

Abstract: Modeling of esterification reaction of ethanol with acetic acid in a reactive batch distillation column is investigated. The dynamic model developed is verified using the data of a theoretical study available in the literature. However, the existing models are found to be inappropriate for this system when compared with the experimental data. Then the model is improved using the data obtained from the experiments performed on a lab-scale column. In the model, different rate expressions and different thermodynamic models ($\phi-\phi$, EOS-G_{ex}, and $\gamma-\phi$ methods) considering different equations of state (EOS), mixing rules and activity coefficient models are used. It is found that the $\gamma-\phi$ approach considering van der Waals mixing rule and NRTL activity coefficient model gives the best fit between the dynamic model and the results of the experiments for the system under study.

Keywords: Reactive Distillation, Batch Column, Mathematical Modeling, Dynamic Simulation, Ethyl Acetate Production.

1. INTRODUCTION

Reactive distillation, which is combination of reaction and separation operations in a single unit, has many advantages over conventional processes. Modeling of this process is a challenging task due to its complex dynamics, highly nonlinear behaviour, complex interactions between vapor-liquid equilibrium (VLE) and chemical kinetics.

The system studied in this work is an esterification reaction where ethanol (EtOH) reacts with acetic acid (AcAc) to produce ethyl acetate (EtAc) and water (H_2O). In this quaternary system, azeotropes are formed between EtOH-H_2O, EtAc-H_2O, EtAc-EtOH, and EtAc-H_2O-EtOH. In the literature, most of the studies on this reaction utilized the numerical methods of solution (Chang and Seader, 1988; Bogacki et al., 1989; Simandl and Svrcek, 1991) and some others worked on its thermodynamics for phase equilibrium (Okur and Bayramoglu, 2001; Park et al., 2006) with very simple models in simulation. Assumptions considered are; ideal plates with constant molar holdup, negligible tray hydrodynamics and steady state condition. Alejski and Duprat (1996) dealt with the dynamic simulation of a reactive distillation column for EtAc system in presence of a catalyst. Tang et al. (2003) showed that, NRTL activity coefficient model parameters predict the VLE data of this system well. Both of these dynamic studies are done on continuous column. On the other hand, unlike continuous columns very few studies are done for modeling of reactive batch columns. Mujtaba and Macchietto (1997) developed an optimization algorithm and Monroy-Loperena and Alvarez-Ramirez (2000) developed an output-feedback control algorithm for a reactive batch column. However, in their studies they used very simplified VLE models and the model is not checked with experimental data.

The objective of this study is to develop a dynamic mathematical model for the esterification reaction of EtOH and AcAc in a reactive batch distillation column (RBDC) by verifying it with experimental data. Thus, different thermodynamic models are used for VLE calculations in order to obtain a good fit with the experimental data.

2. EXPERIMENTAL

The batch distillation column (Fig. 1) used in this study (Bahar, 2007) has an inner diameter of 5 cm, a height of 40 cm, and 8 sieve plates. The overall column parameters and experimental operating conditions are given in Table 1. The column is first operated at total reflux. After steady state is reached, reflux ratio is set to a predefined value. Analyses of the collected samples are done through Gas Chromatography.

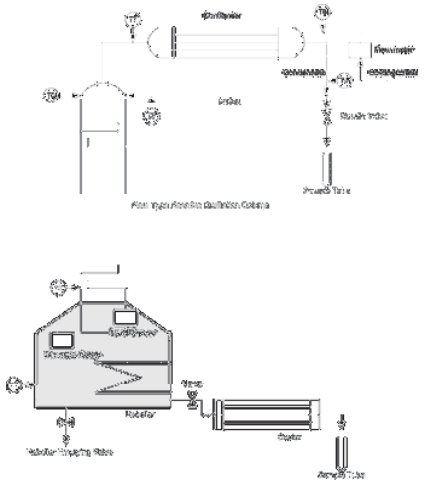


Fig. 1. Reactive Batch Distillation Column.

Table 1. Experimental Column Parameters and Operating Conditions

Total fresh feed, mol	311.67
Feed composition (EtAc, EtOH, H_2O, AcAc), mole fraction	0.0, 0.5, 0.0, 0.5
Column holdup, mol	
condenser+drum	30
internal plates	0.779
Reboiler heat duty, J/h	2.016×10^6
Column pressure, bar	1.013
Cooling water flow rate, lt/min	1.0

3. RBDC MODELING

The unsteady state model of RBDC is based on model of Yıldız et al. (2005). The assumptions employed are negligible vapour holdup, constant volume of tray liquid holdup, constant liquid molar holdup in the reflux drum, total condenser, negligible fluid dynamic lags, linear pressure drop profile, Murphree tray efficiency, approximated enthalpy derivatives and adiabatic operation. The balance equations for column reboiler, trays and reflux-drum-condenser system are given as follows:

Reboiler: j = 1,…,NC

$$dM_1/dt = L_2 - V_1 \qquad (1)$$

$$d(M_1 x_{1j})/dt = L_2 x_{2j} - V_1 y_{1j} + \varepsilon_j R_1 M_1 \qquad (2)$$

$$d(M_1 h_1)/dt = L_2 h_2 - V_1 H_1 + Q_1 \qquad (3)$$

where the reboiler holdup, M_1, is as given in (4) where M_f^0 is the molar amount of feed initially charged to the column.

$$M_1 = M_f^0 - \sum_{n=2}^{NT+2} M_n - \int_0^t D(\tau)d\tau \qquad (4)$$

Trays: i = 2,…,N_{T+1}; j = 1,…,NC

$$dM_i/dt = L_{i+1} + V_{i-1} - L_i - V_i \qquad (5)$$

$$d(M_i x_{ij})/dt = L_{i+1}x_{i+1,j} + V_{i-1}y_{i-1,j} - L_i x_{ij} - V_i y_{ij} + \varepsilon_j R_i M_i \qquad (6)$$

$$d(M_i h_i)/dt = L_{i+1}h_{i+1} + V_{i-1}H_{i-1} - L_i h_i - V_i H_i \qquad (7)$$

where $M_i = (\rho_i^{avg} / Mw_i^{avg})v_i$ is the molar holdups on trays where ρ_i^{avg} is the average density of the mixture on the i[th] tray, Mw_i^{avg} is the average molecular weight of the mixture on the i[th] tray, v_i is the volume of the liquid tray holdup.

Reflux-drum-condenser system: j = 1,…,NC

$$dM_{NT+2}/dt = V_{NT+1} - L_{NT+2} - D \qquad (8)$$

$$\frac{d(M_{NT+2}x_{NT+2,j})}{dt} = V_{NT+1}y_{NT+1,j} - L_{NT+2}x_{NT+2,j} - Dx_{NT+2,j} + \varepsilon_j R_{NT+2} M_{NT+2} \qquad (9)$$

$$\frac{d(M_{NT+2}h_{NT+2})}{dt} = V_{NT+1}H_{NT+2} - L_{NT+2}h_{NT+2} - Dh_{NT+2} - Q_{NT+2} \qquad (10)$$

where R_i is the reaction rate at i[th] stage in mol/h and can be expressed as given in (11) and the rate expression, r_i, without catalyst is expressed as $r = k_1 x_2 x_4 - k_2 x_3 x_1$. The forward and backward reaction rate constants in lt/gmol.min are k_1=29100exp(-7190/T(K)) and k_2=7380exp(-7190/T(K)), respectively (Alejski and Duprat, 1996).

$$R_i = r_i \rho_i / MW_i \quad \text{for i = 1,…,}N_{T+2} \qquad (11)$$

The reflux ratio is defined as $R = L_{NT+2}/D$. The subscripts i and j are for stage and component numbers, respectively. i=1 for reboiler, i=2,.,N_{T+1} for trays and i=N_{T+2} for reflux-drum-condenser unit. The components are numbered in the subscripts as follows: EtAc-1, EtOH-2, H_2O-3, and AcAc-4.

In energy balance equations, no additional term for the heat of reaction is included because, the enthalpies are referred to their elemental state for which the heat of reaction is accounted automatically and thus, no separate term is needed (Mujtaba and Macchietto, 1997). The linear pressure drop profile is given as $P_i = P_1 - i(P_1 - P_{NT+2})/NT$ where P_i is the pressure in i[th] tray, P_1, the pressure in the reboiler and P_{NT+2}, the pressure in the reflux drum.

4. MODELS FOR VAPOR-LIQUID EQUILIBRIUM

In modeling of batch distillation column, the selection of proper thermodynamic model affects the estimation of compositions highly and therefore is very crucial. In simulation studies, four different models are used for phase equilibrium and these models are explained below in detail.

4.1 Model-I:Phase Equilibrium Using VLE data in Literature

VLE data for EtAc-EtOH-H_2O-AcAc system given in Table 2 is taken from literature (Suzuki et al., 1971). This data is utilized in the simulation as a preliminary check.

Table 2. Vapor Liquid Equilibrium Data.

EtAc	log K = -2.3 x 10^3/T + 6.742
EtOH	log K = -2.3 x 10^3/T + 6.588
H_2O	log K = -2.3 x 10^3/T + 6.484
AcAc	K = (2.25 x 10^{-2})/T - 7.812 for T > 347.6 K
	K = 0.001 for T ≤ 347.6 K

4.2 Model-II: Phase Equilibrium Using $\phi - \phi$ Approach

In this approach, Peng Robinson EOS (PR) with van der Waals one-fluid mixing rule is used to calculate the fugacity of species for both liquid and vapor phases. The binary interaction parameters are given in Table 3 (Burgos-Solorzano, 2004).

Table 3. Binary interaction parameters, k_{ij}.

k_{ij}	EtAc	EtOH	H_2O	AcAc
EtAc	0.0	0.022	-0.280	-0.226
EtOH	0.022	0.0	-0.935	-0.0436
H_2O	-0.280	-0.935	0.0	-0.144
AcAc	-0.226	-0.0436	-0.144	0.0

4.3 Model-III: Phase Equilibrium Using Combination of EOS with Excess Free Energy Models (EOS-G_{ex} Approach)

In this method, activity coefficient models are incorporated into EOS. NRTL, Wilson, and UNIQUAC models are used

and performances for the system under consideration are compared. The parameters for these models are obtained from Tang et al. (2003), Okur and Bayramoğlu (2001), and Kang et al., (1992).

In this study, as EOS; PR and Peng-Robinson-Stryjek-Vera (PRSV) (Stryjek and Vera, 1986) are used. κ_1 parameters for components are given in Table 4 (Stryjek and Vera, 1986). As the mixing rule, van der Waals one-fluid mixing rule, Huron-Vidal (Original) Mixing Rule (HVO), and Orbey-Sandler modification of the Huron-Vidal mixing rule (HVOS) are used.

Table 4. PRSV EOS parameters, κ_1 .

Components	κ_1
EtAc	0.0693
EtOH	-0.03374
H_2O	-0.06635
AcAc	-0.19724

4.4 Model-IV: Phase Equilibrium Using $\gamma-\phi$ Approach

In VLE descriptions with the $\gamma-\phi$ approach, an activity coefficient model can be used for the liquid phase and an EOS is used for the vapor phase.

5. RESULTS AND DISCUSSION

This study is done in three phases. In first phase, modeling studies are done and then checked with a simulation study found from the literature which has the same reactive system. In second phase, experimental studies are done and data is collected for total and different reflux ratios. In third phase, the experimental findings and the simulation results are compared and the dynamic model is finalized by selecting the appropriate thermodynamic model for VLE calculations.

The properties of the column which is used in simulation are given in Table 5. Monroy-Loperena and Alvarez-Ramirez (2000) used the VLE data of Model-I and temperature independent rate constants with k_1 of $4.76x10^{-4}$ lt/(gmol.min) and k_2 of $1.63x10^{-4}$ lt/(gmol.min). The comparison of dynamic model using same rate expression at total reflux is given in Fig. 2. As can be seen, the results are almost the same. This indicates that the developed dynamic model is quite satisfactory to represent this non-linear and complex problem of RBDC behaviour.

Table 5. RBDC Specifications

No. of stages (including reboiler and total condenser)	10
Total fresh feed, kmol	5.0
Feed composition (EtAc, EtOH, H_2O, AcAc), mole fraction	0.0, 0.45, 0.1, 0.45
Column holdup, kmol	
condenser	0.1
internal plates	0.0125
Condenser vapor load, kmol/h	2.5
Column pressure, bar	1.013

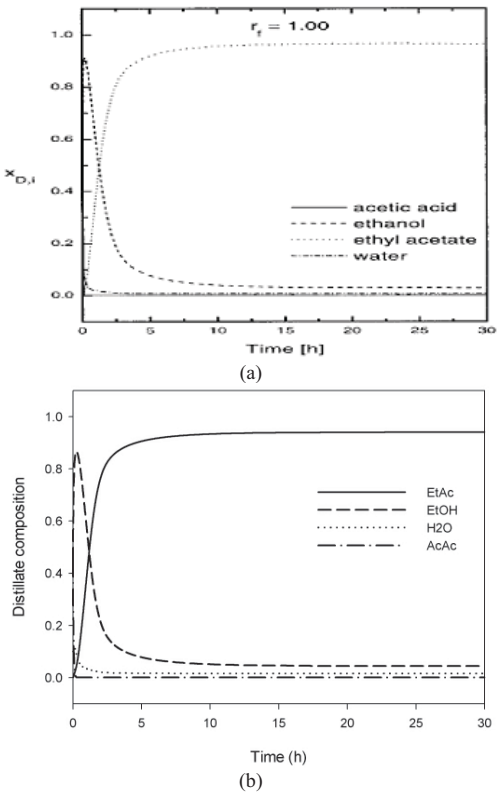


Fig. 2. Distillate compositions at total reflux. (a) Results from literature (b) Results from the simulation in this study.

After obtaining similar results with literature, experiments are performed in order to improve the model. Column is first operated at total reflux until the steady state is reached. Then it is operated with arbitrary reflux ratios and data for distillate and reboiler compositions are collected with respect to time.

5.1 Dynamic Analysis of the Results of Experimental and Simulation Studies

There is a difference in initialization of the experiments and simulation program. Therefore, although the trends of the profiles of compositions are similar, they cannot be compared up to steady state point. Consequently, if the two results match at total reflux steady state, then comparisons of dynamic response can be done. Thus, for each model explained in Section 4, the experimental data collected is checked with the simulation results.

Model-I: In Table 6, simulation result of Model-I and experiments at total reflux steady state are given. It can be seen that, when steady state values are compared, they are too different from each other and there is no need to check the dynamic behaviour. More accurate VLE model is needed.

Table 6. Total Reflux Steady State Composition Values

Comp.	Distillate		Reboiler	
	Exp.	Sim.	Exp.	Sim.
EtAc	0.5222	0.9384	0.1434	0.2582
EtOH	0.2408	0.0476	0.3189	0.1829
H_2O	0.2371	0.0135	0.1918	0.3684
AcAc	0.0000	$5.61x10^{-4}$	0.3459	0.1906

Model-II: The results of simulation that uses Model-II and experiments are given in Fig. 3. It is found that total reflux steady state values are better compared to Model-I. The comparison is further continued dynamically for a constant reflux ratio of 5.72. The time at total reflux steady state is shown as zero. It can be seen that there are great differences in the distillate and reboiler liquid composition trends with respect to time.

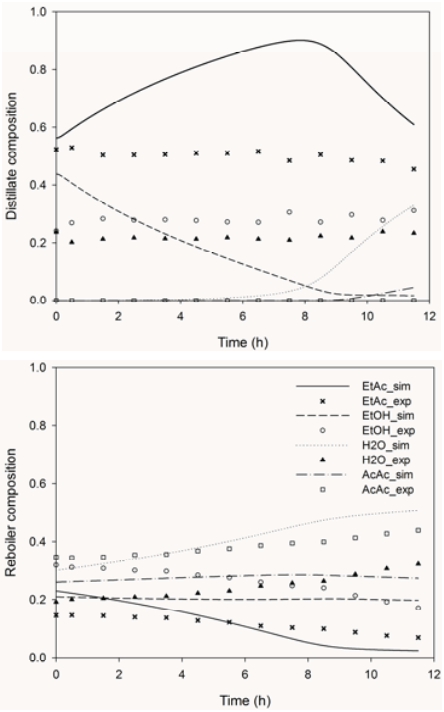


Fig. 3. Results with Model-II

Model-III: In Model-III, first of all PR with HVO mixing rule and NRTL activity coefficient model (Model-III-A) is tested. It can be seen from Fig. 4 that the results are somewhat improved compared to Model-II, especially for the reboiler compositions. However, the results for distillate compositions are not satisfactory. Therefore, EOS is changed to PRSV with same mixing rule and activity coefficient model; the performance of the system with this model (Model-III-B) is given in Fig. 5. Distillate compositions are much better than that of Model-III-A. However, reboiler compositions become worse and therefore this result is also found to be not satisfactory. As a further step, mixing rule is changed to HVOS and it is used together with PRSV and NRTL activity coefficient model (Model-III-C). The results are given in Fig. 6. The results for both distillate and reboiler compositions are improved significantly with this thermodynamic model.

In order to see the effects of different activity coefficient models, Wilson and UNIQUAC models are used in EOS-G_{ex} approach. The distillate and reboiler liquid compositions with Wilson model (Model-III-D) and UNIQUAC model (Model-III-E) are given in Fig. 7 and Fig. 8, respectively. It can be seen from the figures that, while NRTL and Wilson models give similar results, UNIQUAC performs poorly. NRTL model is selected to be the most proper activity coefficient model for this system, since it gives slightly better results than Wilson model, and will be used also in Model-IV.

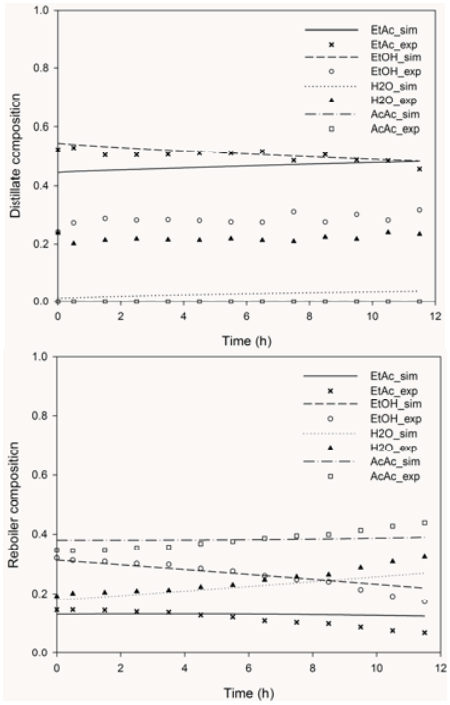


Fig. 4. Results with Model-III-A

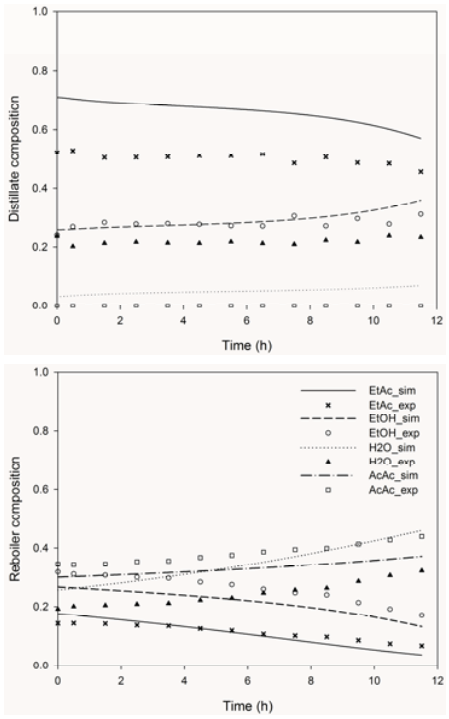


Fig. 5. Results with Model-III-B

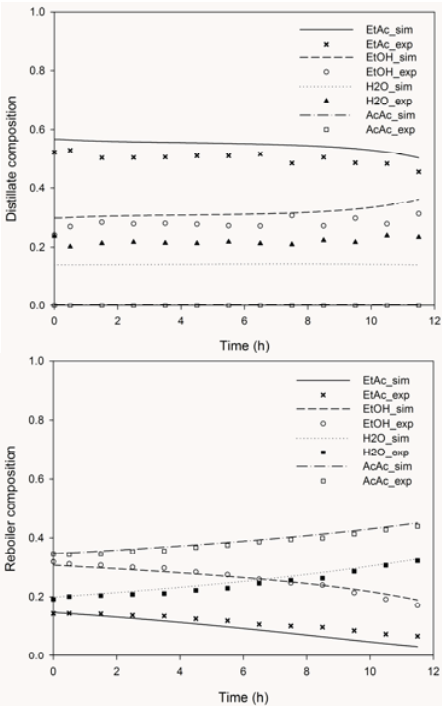


Fig. 6. Results with Model-III-C

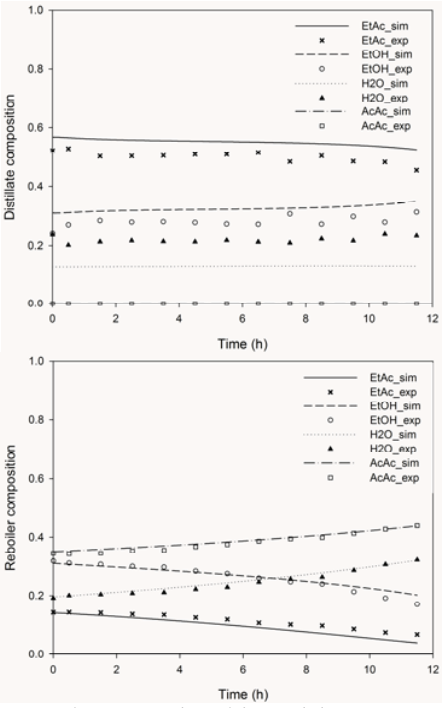


Fig. 7. Results with Model-III-D

Model-IV: In Model-IV, NRTL activity coefficient model is used for liquid phase, PRSV (Model-IV-A) and PR (Model-IV-B) with van der Waals mixing rule is used for vapor phase. It can be seen from Fig. 9 that the distillate compositions are improved compared to Model-III. Although the reboiler compositions become a little worse, they are in an acceptable range. Unlike Model-III, PR also gives similar results with PRSV in Model-IV as can be seen in Fig. 10.

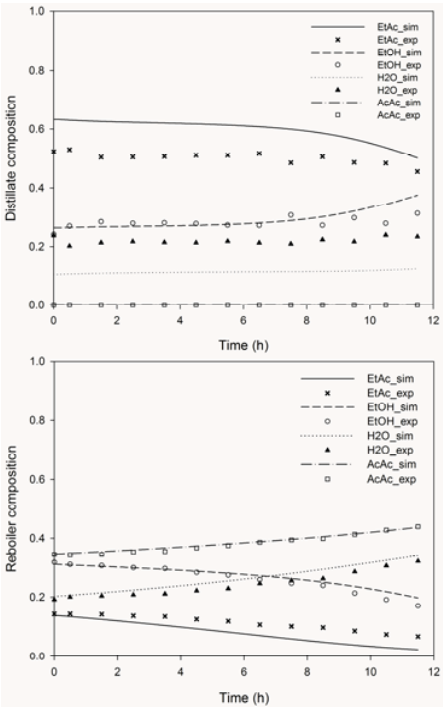


Fig. 8. Results with Model-III-E

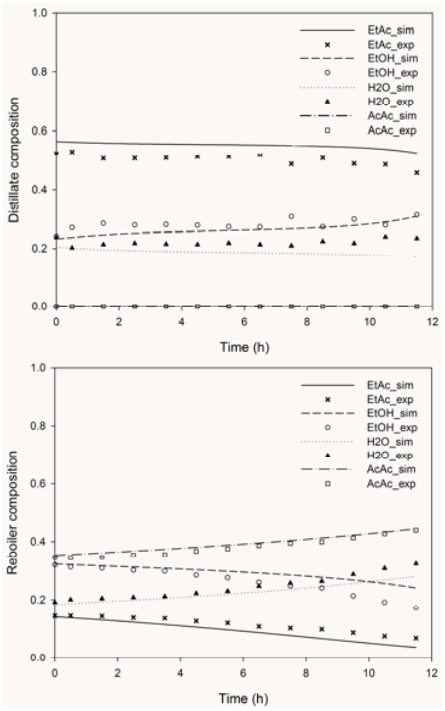


Fig. 9. Results with Model-IV-A

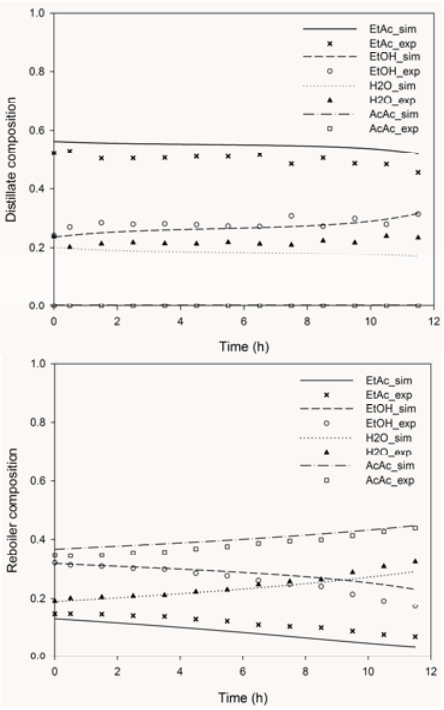


Fig. 10. Results with Model-IV-B

Table 7. Summary of Thermodynamic Models.

Model	*Description*	*IAE Scores*		
		Distillate	*Reboiler*	*Overall*
Model-I	VLE data from literature	-	-	-
Model-II	$\phi-\phi$ method (PR+van der Waals)	-	-	-
Model-III-A	EOS-G^{ex} method (PR + HVO + NRTL)	6.080	1.049	7.129
Model-III-B	EOS-G^{ex} method (PRSV + HVO + NRTL)	4.514	2.805	7.320
Model-III-C	EOS-G^{ex} method (PRSV+HVOS+NRTL)	2.131	0.621	2.751
Model-III-D	EOS-G^{ex} method (PRSV+HVOS+Wilson)	2.437	0.552	2.989
Model-III-E	EOS-G^{ex} method (PRSV+HVOS+UNIQUAC)	2.915	0.877	3.791
Model-IV-A	$\gamma-\phi$ method (PRSV+van der Waals+NRTL)	1.321	1.026	2.347
Model-IV-B	$\gamma-\phi$ method (PR+van der Waals+NRTL)	1.279	1.072	2.351

6. CONCLUSIONS

A summary of results for different thermodynamic models are given in Table 7 with Integral Absolute Error (IAE) scores of response curves. Model-IV-A, which uses traditional $\gamma-\phi$ approach with NRTL activity coefficient model for liquid phase and PRSV for vapor phase gives the smallest IAE score for the quaternary EtOH-AcAc-EtAc-H_2O system. Nevertheless, Model-IV-B which uses the traditional approach with NRTL activity coefficient model for the liquid phase and the PR-EOS for the vapor phase, which is simple to use also gives similar result with a slightly higher IAE score. Thus, both methods are suggested to be used in the simulation of EtOH esterification reaction with AcAc in a RBDC system.

REFERENCES

Alejski, K., Duprat, F. (1996). Dynamic Simulation of the Multicomponent Reactive Distillation. Chem Eng Sci, 51 (18), 4237-4252.

Bahar, A. (2007). Modeling and Control Studies for a Reactive Batch Distillation Column, in Ph.D. Thesis, Chemical Engineering Department, Middle East Technical University, Ankara.

Bogacki, M.B., Alejski, K., Szymanowski, J. (1989). The Fast Method of the Solution of a Reacting Distillation Problem. Comput Chem Eng, 13 (9), 1081-1085.

Burgos-Solorzano, G.I., Brennecke, J.F., Stadtherr, M.A. (2004). Validated Computing Approach for High-Pressure Chemical and Multiphase Equilibrium. Fluid Phase Equilibr, 219 (2), 245-255.

Chang, Y.A., Seader, J.D. (1988). Simulation of Continuous Reactive Distillation by a Homotopy-Continuation Method. Comput Chem Eng, 12 (12), 1243-1255.

Kang, Y. W., Lee, Y. Y., Lee, W. K. (1992). Vapor-Liquid Equilibria with Chemical Reaction Equilibrium - Systems Containing Acetic Acid, Ethly Alcohol, Water, and Ethyl Acetate. J Chem Eng Jpn, 25 (6), 649-655.

Monroy-Loperena R., Alvarez-Ramirez, J. (2000), Output-Feedback Control of Reactive Batch Distillation Columns. Ind Eng Chem Res, 39, 378-386.

Mujtaba, I.M., Macchietto, S. (1997). Efficient Optimization of Batch Distillation with Chemical Reaction Using Polynomial Curve Fitting Techniques. Ind Eng Chem Res, 36, 2287-2295.

Okur, H. and Bayramoglu, M. (2001), The Effect of the Liquid-Phase Activity Model on the Simulation of Ethyl Acetate Production by Reactive Distillation. Ind Eng Chem Res, 40, 3639-3646.

Park J., Lee, N., Park, S., Cho, J. (2006). Experimental and Simulation Study on the Reactive Distillation Process for the Production of Ethyl Acetate. J Ind Eng Chem, 12(4), 516-521.

Simandl, J. Svrcek, W.Y. (1991). Extension of the Simultaneous-Solution and Inside-Outside Algorithms to Distillation with Chemical Reactions. Comput Chem Eng, 15 (5), 337-348.

Stryjek R., Vera J.H. (1986). An Improved Peng-Robinson Equation of State for Pure Compounds and Mixtures, 64 (2), 323-333.

Suzuki, I., Yagi, H., Komatsu, H., Hirata. M. (1971). Calculation of Multi-component Distillation Accompanied by a Chemical Reaction. J Chem Eng Jpn, 4 (1), 26-32.

Tang, Y. T., Huang, H., Chien, I. (2003). Design of a Complete Ethyl Acetate Reactive Distillation System. J Chem Eng Jpn, 36 (11), 1352-1363.

Yıldız, U., Gürkan, U.A., Özgen, C., Leblebicioğlu, K. (2005), State Estimator Design for Multicomponent Batch Distillation Columns. Chem Eng Res Des, 83(A5), 1-12.

Process Control Applications

Poster Session

Application of the IHMPC to an industrial process system

O. L. Carrapiço[1], M. M. Santos[2], A. C. Zanin[1] and D. Odloak[3]
1. Petrobras, Refinery of Cubatão, Pça Stênio Caio de Albuquerque Lima,1, 11555-900, Cubatão, SP - Brazil
2. Chemtech, a Siemens Company, Av. Ermano Marchetti, 1435, 05038-001, São Paulo, SP - Brazil
3. Department of Chemical Engineering - University of São Paulo, PO.B. 61548, 05424-970, São Paulo, SP – Brazil

Abstract: This paper addresses the application of a new MPC to a distillation system where isobutane and light butenes are separated from butane and heavier compounds. This system is located in the alkylation unit of an oil refinery. The MPC considered here is based on the infinite horizon MPC extended to the case where the system has stable and integrating modes. The controller is developed based on a particular state space model in the incremental form, which considers the existence of time delays. The proposed controller provides nominal stability to the closed loop system. Practical tests in a distillation system show that the performance of the new controller, which can be extended to consider robustness to model uncertainty is similar to the performance of the conventional MPC with finite prediction horizon.

Keywords: Predictive control; Stability; Infinite horizon; Integrating system.

1. INTRODUCTION

One of the key issues in the application of MPC to industrial processes is the requirement that the closed loop system should remain stable for a large set of tuning parameters and any possible control structure in terms of active controlled outputs and available manipulated inputs. Rawlings & Muske (1993) have demonstrated that, in the regulator operation of stable systems, the infinite horizon MPC preserves stability even in the presence of constraints in the inputs and states. These ideas have been extended to the case of output tracking of stable systems (Odloak, 2004) and to systems with stable and integrating modes (Carrapiço & Odloak, 2005; González et al., 2007). However, a recent review by Qin & Badgwell (2003) points out that these developments have not been incorporated into the available MPC technology. Thus, the main scope of this work is to report the application of an infinite horizon MPC with nominal stability to an industrial system of small dimension but that presents the typical ingredients of a practical application: time delay, measured and unmeasured disturbances and integrating modes.

The state space model considered here is an extension of the model developed by Gouvêa & Odloak (1997) and Rodrigues and Odloak (2003) and implemented by Porfírio et al. (2003) to include time delays and integrating modes and is represented as follows:

$$\begin{aligned} x(k+1) &= Ax(k) + B\Delta u(k) \\ y(k) &= Cx(k) \end{aligned} \tag{1}$$

where

$$x(k) = \left[y(k|k)^T \quad y(k+1|k)^T \quad \cdots \quad y(n+np|k)^T \quad x^s(k)^T \quad x^d(k)^T \quad x^i(k)^T \right]^T \tag{2}$$

$$A = \begin{bmatrix} 0 & I_{ny} & 0 & \cdots & 0 & 0 & 0 & 0 \\ 0 & 0 & I_{ny} & \cdots & 0 & 0 & 0 & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & 0 & \cdots & I_{ny} & 0 & 0 & 0 \\ 0 & 0 & 0 & \cdots & 0 & I_{ny} & \Psi((np+1)\Delta t) & I^*((np+1)\Delta t) \\ 0 & 0 & 0 & \cdots & 0 & I_{ny} & 0 & I_{ny}\Delta t \\ 0 & 0 & 0 & \cdots & 0 & 0 & F & 0 \\ 0 & 0 & 0 & \cdots & 0 & 0 & 0 & I_{ny} \end{bmatrix} \tag{3}$$

$$B = \left[S(\Delta t)^T \quad S(2\Delta t)^T \quad \cdots \quad S((np+1)\Delta t)^T \quad [D^0 + \Delta t\, D^i]^T \quad [D^d F\ N]^T \quad [D^i]^T \right]^T$$

$$C = \left[I_{ny} \quad 0 \quad \ldots \quad 0 \right]$$

In the model defined in (1), $u \in \Re^{mu}$ is the manipulated input. The first *np* components of the state vector defined in (2) correspond to the output predictions computed at time *k* based solely on past control actions and disturbances, x^s are the state components associated with the integrating modes created by the incremental form of the model, x^d are the state components associated with the stable modes of the system and x^i are the state components associated with the integrating modes of the system. To represent systems with time delays, it is assumed that $np \geq m + \text{int}\left\{\max(\theta_{i,j}/\Delta t)\right\}$, where *m* is the control horizon of the MPC, Δt is the sampling time and $\theta_{i,j}$ is the time delay associated with output y_i and input u_j. In the state matrix defined in (3), one has

$$\Psi\left((np+1)\Delta t\right) = \begin{bmatrix} \Phi_1\left((np+1)\Delta t\right) & & & \\ & \Phi_2\left((np+1)\Delta t\right) & & 0 \\ & & \ddots & \\ 0 & & & \Phi_{ny}\left((np+1)\Delta t\right) \end{bmatrix}$$

$$\Phi_i\left((np+1)\Delta t\right) = \begin{bmatrix} f_{i,1,1} & \cdots & f_{i,1,na} & f_{i,2,1} & \cdots & f_{i,2,na} & \cdots & f_{i,nu,1} & \cdots & f_{i,nu,na} \end{bmatrix}$$

where $f_{i,j,g} = r_{i,j,g}^{(np+1)\Delta t - \theta_{i,j}}$, r is a stable pole of the system.

States $y(k/k),\ y(k+1/k),\ ....,\ y(k+np/k)$ correspond to the output predictions calculated at time k based solely on past control actions and disturbances.

$$I^*[(np+1)\Delta t] = \begin{bmatrix} (np+1)\Delta t - \theta_1 & 0 & \cdots & 0 \\ 0 & (np+1)\Delta t - \theta_2 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & (np+1)\Delta t - \theta_{ny} \end{bmatrix} \in \Re^{ny \times ny}$$

θ_i is the time delay associated with the integrating mode related to output y_i.

$D^0 \in \Re^{ny \times nu}$ and $D^i \in \Re^{ny \times nu}$

$$F = \text{diag}\left(r_{1,1,1} \cdots r_{1,1,na} \cdots r_{1,nu,1} \cdots r_{1,nu,na} \cdots r_{ny,1,1} \cdots r_{ny,1,na} \cdots r_{ny,nu,1} \cdots r_{ny,nu,na}\right),$$

$F \in C^{nd \times nd}$

$$D^d = \text{diag}\left(d^d_{1,1,1} \cdots d^d_{1,1,na} \cdots d^d_{1,nu,1} \cdots d^d_{1,nu,na} \cdots d^d_{ny,1,1} \cdots d^d_{ny,1,na} \cdots d^d_{ny,nu,1} \cdots d^d_{ny,nu,na}\right),$$

$D^d \in C^{nd \times nd}$

$$N = \begin{bmatrix} J_1 \\ J_2 \\ \vdots \\ J_{ny} \end{bmatrix},\ N \in \Re^{nd \times nu};\ J_i = \begin{bmatrix} 1 & 0 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 1 & 0 & \cdots & 0 \\ & & \ddots & \\ 0 & 0 & \cdots & 1 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & 1 \end{bmatrix},$$

$J_i \in \Re^{nu\ na \times nu}$, $i = 1, 2, \ldots, ny$

In the above matrix, it is assumed that either output y_i integrates only input u_i, or if this output integrates other inputs, the time delays between the integrated inputs and the output are the same for all the inputs. If this condition is not satisfied, the model proposed above is not observable. The step response of the system can be calculated by the following equation

$S(t) = D^0 + \Psi(t) D^d N + I^*(t) D^i$

where t is supposed to be larger than any time delay included in the process model.

2. THE INFINITE HORIZON MPC

The infinite horizon MPC considered here is based on the following control cost

$$V_k = \sum_{j=0}^{\infty} \left[e(k+j|k) - \delta_k^s - j\Delta t \delta_k^i\right]^T Q \left[e(k+j|k) - \delta_k^s - j\Delta t \delta_k^i\right] + \sum_{j=0}^{m-1} \Delta u(k+j|k)^T R\, \Delta u(k+j|k) + \delta_k^{sT} S_1 \delta_k^s + \delta_k^{iT} S_2 \delta_k^i \quad (4)$$

where

$e(k+j|k) = \tilde{y}(k+j|k) - y^{sp}$ and $\tilde{y}(k+j|k)$ is the output prediction at time $k+j$ computed at time k and considering the future control actions. Weight matrices Q, R, S_1 and S_2 are assumed positive definite.

The control objective defined in (4) can be expanded as follows

$$V_k = V_k^{(1)} + V_k^{(2)} + \sum_{j=0}^{m-1} \Delta u(k+j|k)^T R\, \Delta u(k+j|k) + \delta_k^{sT} S_1 \delta_k^s + \delta_k^{iT} S_2 \delta_k^i \quad (5)$$

where

$$V_k^{(1)} = \sum_{j=0}^{np} \left[e(k+j|k) - \delta_k^s - j\Delta t \delta_k^i\right]^T Q \times \left[e(k+j|k) - \delta_k^s - j\Delta t \delta_k^i\right]$$

$$V_k^{(2)} = \sum_{j=1}^{\infty} \left[e(k+np+j|k) - \delta_k^s - (np+j)\Delta t \delta_k^i\right]^T Q \times \left[e(k+np+j|k) - \delta_k^s - (np+j)\Delta t \delta_k^i\right] \quad (6)$$

It can be shown that

$$V_k^{(1)} = \left[\bar{C}\, x(k|k) + \tilde{C}\Delta u_k - y_1^{sp} - \bar{I}\,\delta_k^s - \tilde{I}\,\delta_k^i\right]^T \bar{Q} \times \left[\bar{C}\, x(k|k) + \tilde{C}\Delta u_k - y_1^{sp} - \bar{I}\,\delta_k^s - \tilde{I}\,\delta_k^i\right] \quad (7)$$

where

$$\bar{C} = \begin{bmatrix} C \\ CA \\ \vdots \\ CA^{np} \end{bmatrix},\ \tilde{C} = \begin{bmatrix} 0 & 0 & 0 & \cdots & 0 \\ CB & 0 & 0 & \cdots & 0 \\ CAB & CB & 0 & \cdots & 0 \\ \vdots & \vdots & \vdots & \cdots & \vdots \\ CA^{np-1}B & CA^{np-2}B & & \cdots & CA^{np-m}B \end{bmatrix},$$

$\Delta u_k = \left[\Delta u(k|k)^T \quad \Delta u(k+1|k)^T \quad \cdots \quad \Delta u(k+m-1|k)^T\right]^T$

$\bar{Q} = diag(\left[Q \quad Q \quad ... \quad Q\right]) \in \Re^{np.ny \times np.ny}$,

$y_1^{sp} = [y^{spT} \quad y^{spT} \quad ... \quad y^{spT}]^T \in \Re^{np.ny}$

$\bar{I} = [I_{ny} \quad I_{ny} \quad ... \quad I_{ny}]^T \in \Re^{np.ny\ x\ ny}$,

$\tilde{I} = [0 \quad \Delta t\, I_{ny} \quad ... \quad np\Delta t\, I_{ny}]^T \in \Re^{np.ny\ x\ ny}$

In order to develop the infinite sum defined in (6), one needs to consider an expression for the calculation of the output prediction at time steps beyond time np. Using the model expressions defined in (1) to (3), the output prediction at time step np+1 can be written as follows:

$$\tilde{y}(k+np+1|k) = \tilde{x}^s(k+m|k) + \Psi((np+1)\Delta t)\tilde{x}^d(k+m|k) + (np+1)\Delta t\, I_{ny}\tilde{x}^i(k+m|k)$$

where $\tilde{x}^s$, $\tilde{x}^d$ and $\tilde{x}^i$ are computed considering the future control actions. Analogously, the prediction at any time step $np+j$ can be written as follows:

$$\tilde{y}(k+np+j|k) = \tilde{x}^s(k+m|k) + \Psi((np+1)\Delta t)F^{j-1}\tilde{x}^d(k+m|k) + (np+j)\Delta t\ \tilde{x}^i(k+m|k)$$

In order to guarantee that $V_k^{(2)}$ will be bounded, it is necessary to force the state components related to the integrating modes to be zero at the end of the control horizon:

$$\tilde{x}^s(k+m|k) - y^{sp} - \delta_k^s = 0 \quad (8)$$

$$\tilde{x}^i(k+m/k) - \delta_k^i = 0 \quad (9)$$

If constraints (8) and (9) are satisfied, $V_k^{(2)}$ can be written as follows:

$$V_k^{(2)} = \sum_{j=1}^{\infty} \tilde{x}^d(k+m/k)^T (F^{j-1})^T \left(\Psi((np+1)\Delta t)\right)^T Q \times \Psi((np+1)\Delta t)F^{j-1}\tilde{x}^d(k+m/k)$$

$$V_k^{(2)} = \tilde{x}^d(k+m/k)^T \tilde{Q}\, \tilde{x}^d(k+m/k) \quad (10)$$

where $\tilde{Q}$ satisfies

$$F^T\tilde{Q}F - \tilde{Q} = \left(\Psi((np+1)\Delta t)\right)^T Q\Psi((np+1)\Delta t)$$

and

$$x^d(k+m/k) = F^m x^d(k/k) + F_u \Delta u_k$$

$$F_u = \begin{bmatrix} F^{m-1}B^d & F^{m-2}B^d & \cdots & B^d \end{bmatrix}, \quad B^d = D^d F\, N$$

Substituting (7) and (10) in (4), the control objective becomes

$$V_k = \begin{bmatrix} \Delta u_k^T & \delta_k^{sT} & \delta_k^{iT} \end{bmatrix} H \begin{bmatrix} \Delta u_k \\ \delta_k^s \\ \delta_k^i \end{bmatrix} + 2C_f^T \begin{bmatrix} \Delta u_k \\ \delta_k^s \\ \delta_k^i \end{bmatrix}_k + c \quad (11)$$

where

$$H = \begin{bmatrix} H_{11} & H_{12} & H_{13} \\ H_{12}^T & H_{22} & H_{23} \\ H_{13}^T & H_{23}^T & H_{33} \end{bmatrix}$$

$$C_f = \begin{bmatrix} \left[x(k/k)^T \bar{C}^T\bar{Q}\tilde{C} - y_1^{spT}\bar{Q}\tilde{C} + x^d(k/k)^T F^{mT}\tilde{Q}F_u\right]^T \\ \left[-x(k/k)^T \bar{C}^T\bar{Q}\bar{I} + y_1^{spT}\bar{Q}\bar{I}\right]^T \\ \left[-x(k/k)^T \bar{C}^T\bar{Q}\tilde{I} + y_1^{spT}\bar{Q}\tilde{I}\right]^T \end{bmatrix}$$

$$H_{11} = \tilde{C}^T\bar{Q}\tilde{C} + F_u^T\tilde{Q}F_u + \bar{R}$$

$$c = x(k/k)^T\bar{C}^T\bar{Q}\bar{C}x(k/k) - 2y_1^{spT}\bar{Q}\bar{C}x(k/k) + y_1^{spT}\bar{Q}y_1^{sp} + x^d(k/k)^T F^{mT}\tilde{Q}F^m x^d(k/k)$$

$$\bar{R} = diag(\begin{bmatrix} R & R & \ldots & R \end{bmatrix}) \in \Re^{m.nu \times m.nu}$$

$$H_{12} = -\tilde{C}^T\bar{Q}\bar{I}, \quad H_{13} = -\tilde{C}^T\bar{Q}\tilde{I}, \quad H_{22} = \bar{I}^T\bar{Q}\bar{I} + S_1,$$
$$H_{23} = \bar{I}^T\bar{Q}\tilde{I}, \quad H_{33} = \tilde{I}^T\bar{Q}\tilde{I} + S_2$$

It is easy to show that for the model defined in (1), the constraint defined in (8) can be written as follows

$$x^s(k/k) - y^{sp} + m\Delta t\, x^i(k/k) + \tilde{D}\Delta u_k - \delta_k^s = 0 \quad (12)$$

where

$$\tilde{D} = \begin{bmatrix} D^0 + m\Delta t\, D^i & D^0 + (m-1)\Delta t\, D^i & \ldots & D^0 + \Delta t\, D^i \end{bmatrix}$$

Analogously, the constraint defined in (9) becomes

$$x^i(k/k) + \tilde{D}^i\, \Delta u_k - \delta_k^i = 0 \quad (13)$$

where

$$\tilde{D}^i = \left[\underbrace{D^i \quad \cdots \quad D^i}_{m}\right]$$

Therefore, the infinite horizon MPC that is implemented here is based on the following optimization problem:

$$\min_{\Delta u_k, \delta_k^s, \delta_k^i} V_k \quad (14)$$

subject to

(12), (13) and

$\Delta u(k+j) \in \mathbb{U},\ j \geq 0$,

where

$$\mathbb{U} = \left\{ \Delta u(k+j) \left| \begin{array}{l} -\Delta u^{\max} \leq \Delta u(k+j) \leq \Delta u^{\max} \\ \Delta u(k+j) = 0; \quad j \geq m \\ u^{\min} \leq u(k-1) + \sum_{i=0}^{j} \Delta u(k+i) \leq u^{\max}; \\ j = 0,1,\cdots,m-1 \end{array} \right. \right\}$$

Carrapiço & Odloak (2005) showed that the problem defined in (14) produces a nominally stable MPC if it is solved in a two step approach. In the first step, the objective is to minimize slack δ_k^i, which is related to the integrating modes. Then, in the second step, the objective is to minimize V_k while δ_k^i is kept at the same value computed in the first step.

In practical terms, the two step approach would be equivalent to adopting the value of the slack weight S_2 large enough to force the controller to minimize δ_k^i before considering the other control objectives. Thus, the IHMPC, which is implemented here, is obtained through the solution to the problem defined in (14) with suitable tuning parameters that produce the nominal stability of the controller.

3. PROCESS OVERVIEW AND CONTROL STRATEGY

A schematic representation of the de-isobutanizer distillation column where the infinite horizon MPC was implemented is illustrated in Figure 1. The system is part of an alkylation unit in the PETROBRAS/Cubatão oil refinery. The feed stream distillation column comes from the FCC unit and consists of a mixture of isobutane, 1-butene, cis-2-butene, trans-2-butene, n-butane and n-pentane. The top product, which is sent to the alkylation reactor, is composed mainly of isobutane and light butenes. The bottom stream that is

composed of n-butane and heavy butenes is sent to storage and sold as a special product.

The feed flowrate is defined by the refinery production plan and usually remains constant over long periods of time. The feed temperature is the main disturbances to the control system. A recycle stream of isobutane and steam are used as sources of heat to the reboiler. The pressure in the top drum is controlled by manipulating the bypass of the top condenser. In the original regulatory strategy, a PID controller of the temperature of tray 68 cascades the steam flowrate to the reboiler and there was no control on the level of liquid in the top drum. The main control objective is to keep the composition of the top product composition at desired values.

As shown in Figure 1, in the control strategy implemented in the IHMPC there are two manipulated inputs: u_1 (ton/h) is the steam flowrate to the reboiler and u_2 (m^3/d) is the reflux flowrate. The feed temperature d_1 (ºC) is a measured disturbance. The outputs of the distillation column are: y_1 (%) the level of liquid in the top drum, y_2 (ºC) the temperature of tray 68 and y_3 (%) the percentage of flooding in the column. The flooding is calculated based on the measured values of some variables of the process. The two degrees of freedom (inputs) are used to control the liquid level in the drum at a fixed set-point and the other two outputs are controlled by zone: the column flooding has to be kept below an upper limit and the temperature in tray #68 has to be kept above a minimum value. Two other outputs that will be included in this control strategy in the near future are the volumetric ratio iC4/(∑ olefin components) in the top product and the volumetric fraction of iC4 in the bottom stream.

Step tests were performed in the distillation column and the resulting transfer function model is the following:

$$\begin{bmatrix} y_1(s) \\ y_2(s) \\ y_3(s) \end{bmatrix} = \begin{bmatrix} \frac{2.3}{s} & \frac{-0.7\times10^{-3}}{s} \\ \frac{4.7e^{-7s}}{9.3s+1} & \frac{1.4\times10^{-3}e^{-2s}}{6.8s+1} \\ \frac{1.9e^{-s}}{10.1s+1} & \frac{61\times10^{-3}e^{-3s}}{6.6s+1} \end{bmatrix} \begin{bmatrix} u_1(s) \\ u_2(s) \end{bmatrix} + \begin{bmatrix} \frac{0.2}{s} \\ \frac{0.4e^{-3s}}{11.6s+1} \\ \frac{0.2e^{-3s}}{12.3s+1} \end{bmatrix} d_1(s)$$

During the identification tests, it was observed that the control valve of the reflux flow rate has shown an erratic behavior probably due to stickiness. Although, this problem could be easily repaired, we found it interesting to perform the evaluation test of the proposed IHMPC in these conditions, as this scenario may be frequently found in industry. So, the sticking reflux control valve becomes an unmeasured disturbance to the MPC controller. The transfer functions represented above relates the outputs to the set points to the regulatory flow control loops.

4. PRACTICAL RESULTS

Figures 2 and 3 show the typical responses of the industrial system with IHMPC when a step disturbance is introduced in the set point of the liquid level. The tuning parameters of the controller are the following: $m = 2$, $\Delta t = 1$, $Q = diag(6,4,1)$, $R = diag(0.1,5)$, $S_1 = S_2 = diag(1,1,1)\times10^3$.

In this case, the column flooding (y_3) was controlled at a fixed set point of 91%, while the temperature in tray #68 (y_2) was kept above the minimum constraint (52ºC). Fig. 3 clearly shows that there is a sticking problem in the valve of the reflux flow rate (u_2), where the process variable (*PV*) has a significant delay in comparison to the corresponding set point (*SV*). The consequence is a continuous cycling of this variable with a period of about 30min. This disturbance is transferred to the controlled variables of the system, but the IHMPC can cope with this situation quite nicely as the amplitude of the resulting oscillation is largely attenuated. Concerning the tuning parameters of the proposed IHMPC, they can be borrowed from the conventional MPC, except the prediction horizon, which is infinite, and the slack weights S_1 and S_2. Typically, these parameters should be two or three orders of magnitude larger than the output weights. The main point related to the slack weights is that they should be large enough to make the hessian matrix H defined in (11) positive definite. If this condition is not satisfied, the integrating outputs may become unbounded or the stable outputs may show offset.

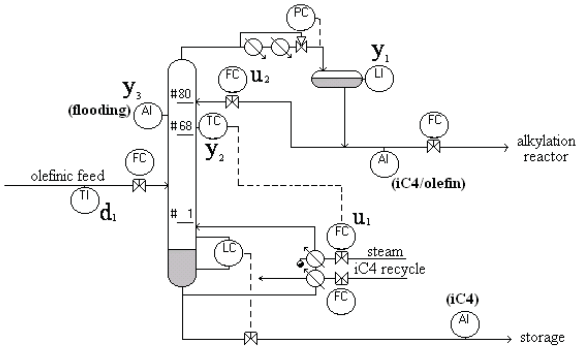


Fig. 1. Schematic diagram of the de-isobutanizer column.

There was a question if the proposed IHMPC would amplify this sort of periodic disturbance, as the controller includes equality constraints (12) and (13) related to cancellation of the integrating modes. Apparently, the inclusion of slacks δ_k^s and δ_k^i greatly reduced this problem. To verify if the gain in stability associated with the use of an infinite prediction horizon would result in a loss of performance, the proposed controller was compared with the conventional MPC. For this purpose, a finite horizon MPC was also implemented in this distillation column. Although, in practice, one cannot repeat exactly the experiment reported above but considering the conventional MPC, Figures 4 and 5 show the responses of the conventional MPC for a similar step disturbance in the set point of the liquid level. The tuning parameters for the two controllers are the same except, for the prediction horizon, which is infinite in the IHMPC and the slacks weights, which do not exist in the conventional MPC. Observing figs. 4 and 5, one may conclude that there is no significant difference between the performances of the two controllers and consequently, there is no practical disadvantage in implementing the infinite horizon MPC that introduces nominal stability.

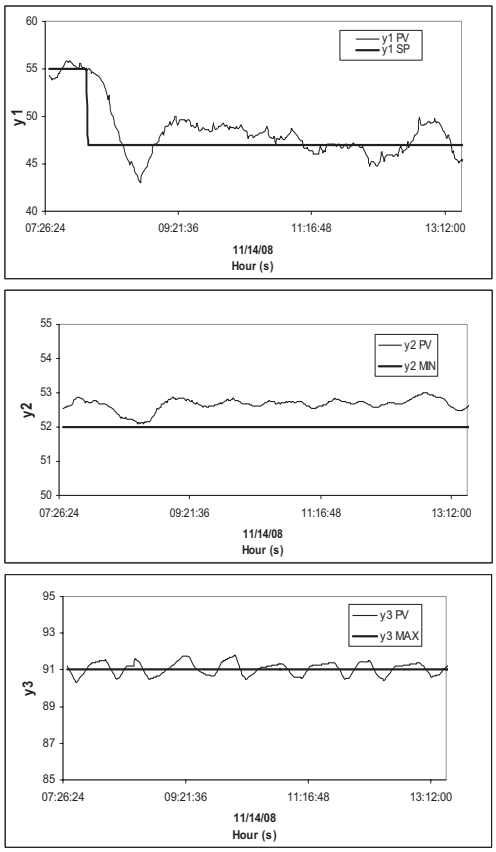


Fig. 2 – Outputs for the IHMPC. Step change in the liquid level set-point.

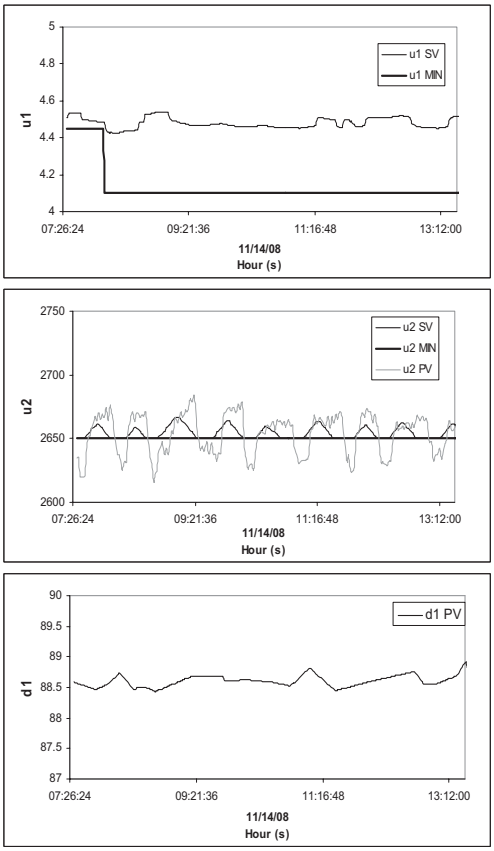


Fig. 3 – Inputs for the IHMPC. Step change in the liquid level set-point.

Another practical experiment performed with the IHMPC is shown in figures 6 and 7. In this case, the set point to the flooding percentage (y_3) in the column is successively decreased along a series of step changes, while the set point to liquid level in the reflux drum (y_1) is fixed and the column temperature (y_2) is controlled by zone. Although the performance of the controller can be considered satisfactory, the sticking problem in the reflux control valve seems more serious and heavily affects the behavior of the system, mainly the reflux flow rate (u_2) and the column flooding. It is not represented here, but the same kind of behavior is observed when the system is controlled with the conventional MPC.

5. CONCLUSIONS

A MPC with infinite prediction horizon was successfully implemented in an industrial distillation column and has been in continuous operation for several months. The proposed controller can be applied to systems with stable and integrating outputs. The IHMPC was compared to the conventional finite horizon MPC and the performances of the two controllers seem quite similar. The new controller has some additional parameters related to the weighting of slack variables that are introduced in the control problem in order to guarantee that this control problem will remain always

REFERENCES

Carrapiço, O. L. & Odloak, D. (2005), A stable model predictive control for integrating processes, *Comp. & Chem. Engng*, 29, 1089–1099.

González, A. H., Marchetti, J. L. & Odloak, D. (2007), Extended robust model predictive control of integrating systems, *AIChE Journal* 53 (7), pp. 1758-1769.

Gouvêa, M. T. & Odloak, D. (1997), Rossmpc: A new way of representing and analysing predictive controllers, *Chem. Engng Research and Design*, 75(7), 693-708.

Odloak, D. (2004), Extended robust model predictive control, *AIChE Journal*, 50(8), 1224-1236.

Porfírio, C. R., Neto, E. A. & Odloak, D. (2003), Multi-model predictive control of an industrial C3/C4 splitter, *Control Engng Practice*, 11 (7), 765-779.

Qin, S. J. & Badgwell, T.A. (2003), A survey of model predictive control technology, *Control Engng Practice* 11(7), 733–764.

Rawlings, J. B., & Muske, K. R. (1993), The stability of constrained receding horizon control, *IEEE Transaction on Automatic Control*, 38, 1512–1516.

Rodrigues, M.A. & Odloak, D. (2003). An infinite horizon model predictive control for stable and integrating processes. *Comp. & Chem. Engng*, 27, 1113-1128.

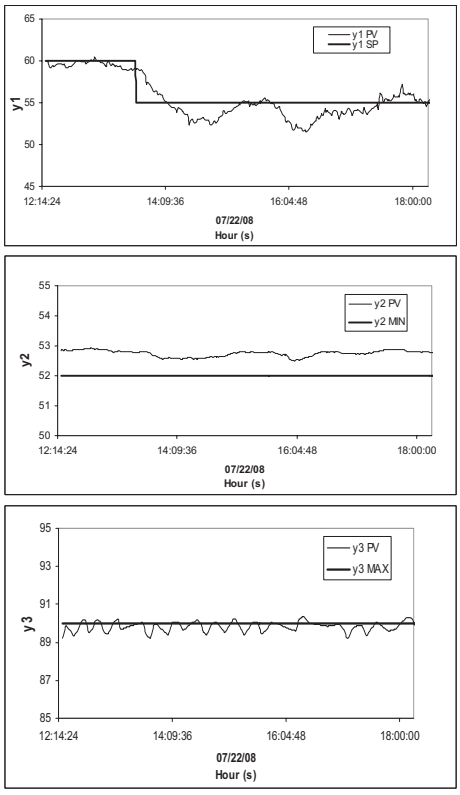


Fig. 4–Outputs for the conventional MPC. Step change in the liquid level set-point.

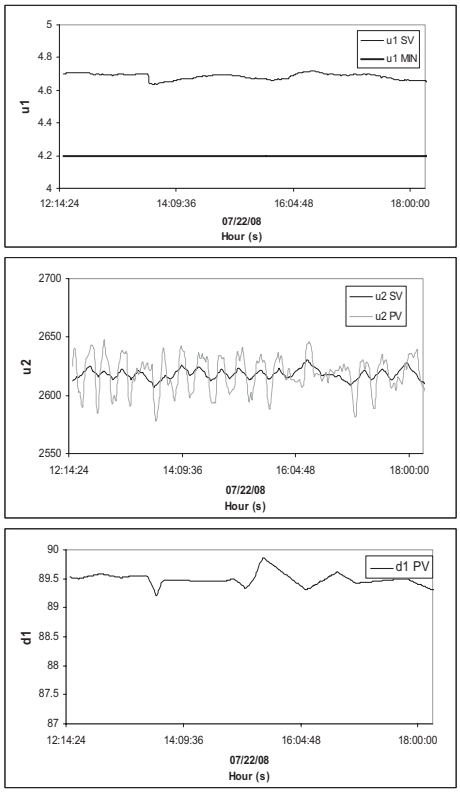


Fig. 5 – Inputs for the conventional MPC. Step change in the liquid level set-point.

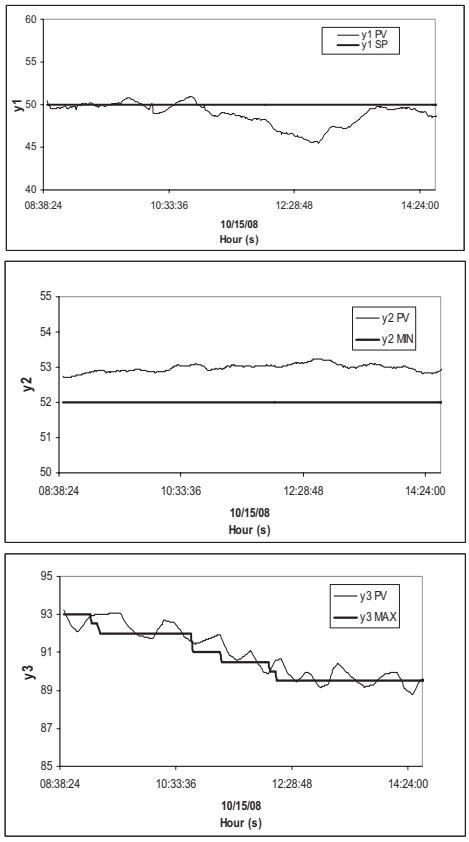


Fig. 6 – Outputs for the IHMPC. Step changes in the set point to column flooding.

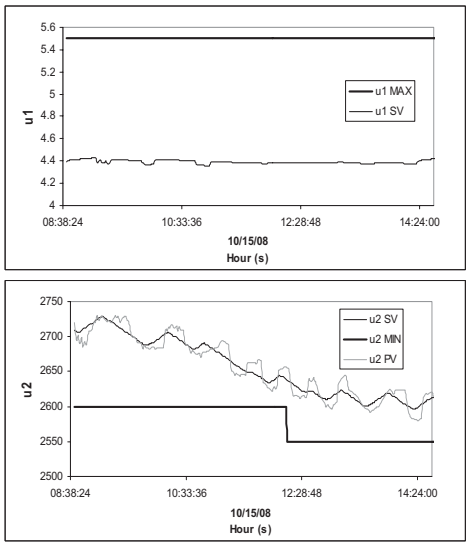


Fig. 7 – Inputs for the IHMPC. Step changes in the set point to column flooding.

Multivariable control with adjustment by decoupling using a distributed action approach in a distillation column

Cintia Marangoni[1]. Joel G. Teleken[2]. Leandro O. Werle[3]. Ricardo A. F. Machado[4]. Ariovaldo Bolzan[5]

Federal University of Santa Catarina State – UFSC . Chemical Engineering Department.
P.O. Box 476 – Trindade, Florianópolis, SC – Zip Code 88010-970 – Brazil. Fone/ Fax: +55 (48) 3721-9554
[1](e-mail:sissi@enq.ufsc.br). [2](e-mail: joel_teleken@yahoo.com.br). [3](e-mail:leandro@enq.ufsc.br).
[4](e-mail:ricardo@enq.ufsc.br) [5](e-mail:abolzan@enq.ufsc.br).

Abstract: This paper presents an approach which uses a multivariable control strategy with distributed action in order to minimize operation transients in distillation columns when a disturbance in the temperature feed is introduced. Experiments were carried out adjusting multivariable PID controllers by static decoupling of the temperature loops of the bottom and distillate trays, characterizing a 2 x 2 system. The dynamics was compared to the distributed approach (same controllers on the bottom and top and an additional control loop on a tray). The controllers adjustment of this new system (3 x 3) was carried out considering the temperature control loop stage in two different ways: decentralized and coupled with the bottom and top temperature control loops. The minimization of transients was verified in both distributed approaches.

Keywords: control schemes, distillation columns, dynamics, multivariable systems, transient analysis.

1. INTRODUCTION

A well designed and adjusted control system is not sufficient to eliminate operation transients of a distillation process. One aspect that contributes to this situation, besides the column operation stage, is the centralization of the control system in the bottom and top column variables. In this way there is the propagation of the corrective control action through the whole unit, generating a production period out of the desired specification. The formation of transients in a distillation column occurs when the process is disturbed and its characteristics reduce the control system efficiency or when an external factor induces the modification of the unit operation point. In the first case there are factors such as variable coupling, nonlinearities, deadtime, high time constants and process constrains. In the second case there are aspects such as the mixture to be distillated, feed composition changes and operation transitions that are necessary due to changes in the market. In both cases, the process dynamics influences the way the transient operation will be generated and what the final result will be.

The current proposals to minimize the transient time of distillation columns use control techniques which consider process dynamics only to study the efficiency of these algorithms, without changing the process conception or evaluating the minimization of transients (Zhu and Liu, 2005). Stricter product specifications and greater demands in terms of environmental control, together with the design of more and more integrated units, require a better performance of these systems. Thus, economic incentives for the development and application of high performance control systems in industrial plants have grown considerably.

The proposal here developed, which is the object of this study, consists of the distribution of the control action throughout the column stages aiming at the minimization of the transient operation. This approach is based on the study of diabatic distillation columns (Koeijer, Rosjorde and Kjestrup, 2005), where intermediate heating points are used instead of only one heat input (reboiler) and one heat remover (condenser). These additional points keep a certain desired temperature profile throughout the column. Previous research (Marangoni and Machado, 2007) has demonstrated the feasibility of this proposal with the use of classic controllers (PID). The unit dynamics was evaluated and the results showed a reduction in the operation transition time when feed disturbances are introduced into the distillation column. Although 90% of industrial processes use classic controllers (Astrom and Hagglun, 2001), it was also necessary to evaluate the use of advanced controllers (model-based) which consider the process dynamics. Multivariable and predictive control seem to be the most used techniques due to their great flexibility. Here, it is worth mentioning the studies carried out with model-based controllers: Model Predictive Control (MPC) (Bezzo et al., 2005); Dynamic Matrix Control (DMC) (Jana et al., 2005); and Generalized Predictive Control (GPC) (Karanca, 2003). On the other hand, some studies have been carried out with proportional-integral-derivative (PID) controllers, aiming at a more flexible adjustment considering the distillation characteristics (Zhu and Liu, 2005). However, even in these recent studies, the controllers used to obtain the quality profile are implemented with the control action only in the bottom and top column stages.

Thus, aiming at the application of easy implementation strategies, the objective of this study was to evaluate the use of a 2 x 2 control system (controllers of the temperature loops

of the bottom and distillate trays) and compare it to a new distributed approach (same controllers on the bottom and top and an additional temperature control loop on a tray) implemented in two different ways: the first considering the additional control stage without interaction with the other two control loops (decentralized), and the second considering the system as 3x3 multivariable.

2. MATERIALS AND METHODS

Experiments were carried out in a pilot unit processing an ethanol-water mixture. The conditions used are summarized in Table 1. Composition measurements were carried out during the experiments using a densimeter for alcohol.

Table 1. Operation conditions used in the experiments.

Variable	Value
Ethanol feed volumetric fraction	0.15
Feed Temperature	92°C
Volumetric feed flow	300 $L.h^{-1}$
Column top pressure	1.25 bar
Drop pressure	0.25 bar
Reflux ratio (Reflux stream/Distillate)	5
Bottom Holdup	4 L
Accumulator Holdup	5 L

2.1 The pilot unit

The unit, illustrated in Figure 1, represents a tray distillation process. It operates in a continuous way and thus there is a main tank responsible for the feed.

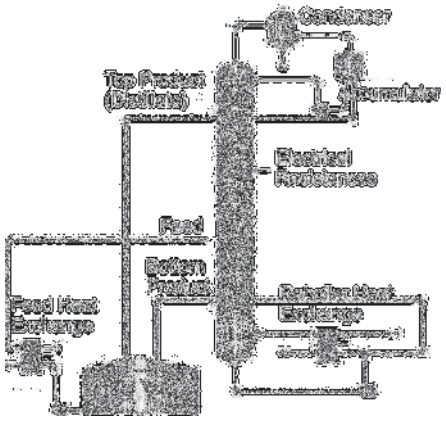


Fig. 1. Schematic illustration of the experimental unit's.

The column has 13 equilibrium stages and each module has one point for temperature measurement, one for sample collection and a third for the distributed heating adaptation. The latter was carried out by means of electrical resistances designed with up to 3.5kW power each. Temperature sensors (Pt-100) were used to monitor this variable in all equilibrium stages, as well as the main tank and the reflux accumulator. The feed was carried out on the fourth tray, with the reboiler as the zero stage.

The control configuration of the distillation column was formulated based on Nooraii et al. (1999), and is illustrated in Figure 2. The following control loops were defined: (1) bottom level control through the bottom product flow rate adjustment; (2) reflux accumulator level control by manipulating the top product flow rate; (3) feed flow rate control as a function of the adjustment of the same stream flow rate; (4) feed temperature control through the fluid flow rate adjustment in the heat exchanger of this stage; (5) last tray (distillate) temperature control by means of the manipulation of the reflux flow rate; (6) reboiler temperature control through the vapor flow rate in the heat exchanger of this stage; and (7) temperature control of pre-defined stages of the column through the adjustment of the dissipated power in the tray electrical resistance.

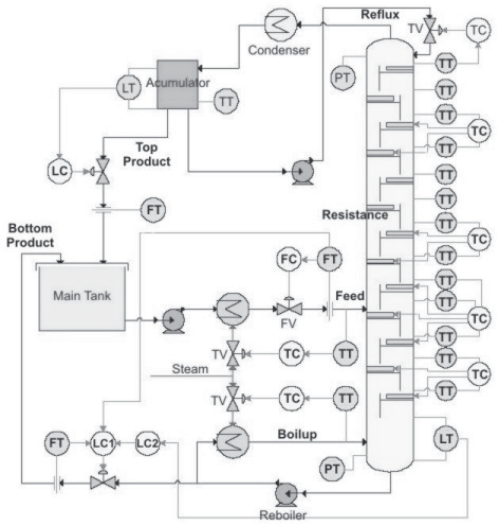


Fig. 2. Control configuration of the distillation unit.

The first, second and third loops represent the column mass balance (inventory) control. The fifth and sixth loops comprise the quality control – in this case represented by the temperature. The use of these two loops in combination is referred to herein as conventional control. When these two loops are combined with the seventh loop mentioned above, it is considered herein as the distributed strategy.

All control loops are instrumented with fieldbus protocols, along with the acquisition and indication of the bottom and distillate stream flows and the pressures at the same stages. The temperatures of all the trays, reboiler, accumulator and feed are monitored by a programmable logic controller and used in the dynamic study of the distributed control. The pressures were monitored in order to assure the proper functioning of the equipment and the process.

2.2 The control strategies tested

For this study, the experiments were carried out with three different control strategies: (1) conventional 2 x 2 – with multivariable control applied to the reboiler and distillate temperatures; (2) distributed 2 x 2 – with multivariable control applied to the reboiler and distillate temperatures and decentralized control in only one stage (PID without interaction with the other loops); and (3) distributed 3 x 3 – with multivariable control applied to the reboiler, second stage and distillate temperatures.

2.3 Controller's tuning

PID controllers are used in the three strategies tested. This kind of controller was employed since it is the most widely used (Astrom and Hagglund, 2001). Multivariable tuning was applied as the experiments consider both loops (reboiler and distillate temperatures) coupled to control the process.

Thus, for strategies 1 and 3 a static decoupler was used (Lee et al, 2005) to cancel the undesired effects of the interaction and adjustment of the multivariable controllers. This procedure can be designed from steady-state process gains, which are easier to obtain and can be tuned in the field. As an initial estimate the PID controller parameters were calculated using the criterion of the integral absolute error (ITAE). A fine adjustment was then made in the plant.

For strategy 2, the same controllers obtained in strategy 1 were used for the reboiler and distillate temperature loops. For the tray temperature controller the ITAE criterion was used to estimate the parameters followed by a fine adjustment (considering this loop decentralized from others, i.e., with weak interaction).

2.4 Stage selection

To identify the most sensitive stage for the consequent application of the distributed control, three different methods were applied (Luyben, 2006). In the first method, the difference between the temperatures of two successive trays was calculated throughout the column and the most sensitive tray was that which presented the greatest difference in relation to its adjacent tray. In the second method, a temperature profile for a given value of the manipulated variable (in this case, the reflux flow and the reboiler heat) is obtained. The most sensitive tray gives a symmetrical response to positive and negative equally variations. Finally, the third method analyzes the tray with the highest derivative of the temperature in relation to the stage when the process is disturbed. To analyze the first method, the temperature profile for three different conditions of ethanol feed composition (15, 25 and 35%) was observed. In the second and third methods it was necessary to disturb the process and evaluate its behavior. The feed flow used was $400L.h^{-1}$ as the standard condition, which was increased to $600L.h^{-1}$ and also decreased to $200L.h^{-1}$.

It is important to emphasize that the different methods can produce different answers. The definition was based on this analysis together with the characteristics of the plant.

2.5 Disturbances

To analyze the control strategies, changes in the temperature feed were introduced, decreasing this variable by around 15°C (from 92°C to 77°C). This was achieved by controlled cooling of this stream.

This study aimed to interfere in the column temperature profile in order to minimize the response time when some disturbance occurs. Thus, it was not tested for set point tracking.

3. RESULTS

The first step of this study was to determine the stage were distributed heating could be applied. As cited before, this was achieved through a sensitivity analysis employing three different methods. The results obtained with the first method (successive trays) using three feed ethanol composition conditions demonstrated the possibility of using trays 1, 2, 3, 5 and 7. As the fifth and seventh trays are located in the rectifying section, they were discarded. It was assumed, following diabatic studies upon which this proposal was based, that in this section it is better to remove heat than supply it.

In addition, as this is an initial study, it was defined that only one tray will be used to test the proposal. To define this stage, since method 1 was not conclusive, the analysis of symmetrical response and maximum derivative (methods 2 and 3) was used. The derivative method again pointed to stages 5 and 7, which were previously discarded, but the symmetrical response method indicated tray 2 as the most appropriate for this study. Figure 3 shows this analysis, where it can be observed that tray 2 is almost the same distance from steady state when the process is disturbed with positive and negative perturbations in the feed flow.

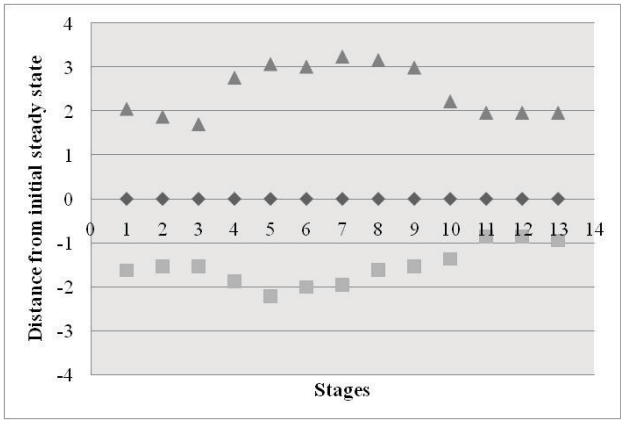


Fig. 3. Results of sensitivity analysis using symmetrical response method (■ negative disturbance, ◆ steady state, ▲ positive disturbance).

Based on this sensitivity analysis, the distributed action of the proposal was used only in tray 2. The simultaneous action of the trays was not tested because the main objective was to analyze the distributed proposal with a multivariable system, and its behavior, through the coupling of control loops.

In sequence, the process transfer functions were determined and the relative gain array matrix was evaluated.

Multivariable control algorithms were studied since the process has multiple inputs and outputs and it is characterized by the high coupling degree among the variables. These kinds of controllers require process models for which an approximated model was used, obtained by the transfer

functions of the reboiler, second stage and distillate temperature control loops. In fact, a multivariable system can be easily modeled through transfer functions. These functions associate the system outputs (Y) with the disturbances (L) and the inputs (U), and integrate the transfer function matrix with the disturbance (G_L) and inputs (G).

The transfer functions were obtained by means of experimental tests, through input and output data collection and later numeric treatment of this information, disturbing the variables that are used for the manipulation of the control loops. The equations obtained are presented below (time values expressed in seconds and deadtime obtained by Taylor series approximation).

Equation (1) presents the matrix which represents the 3 x 3 system, where Tb is the reboiler temperature, Td the distillate temperature and T_2 the second stage temperature, corresponding to the system outputs. Q_b (steam valve opening at the reboiler entrance), R (reflux flow valve opening) and Q_2 (dissipated power at the electrical resistance stage), represent the inputs.

$$\begin{bmatrix} Y_1 = Tb(s) \\ Y_2 = Td(s) \\ Y_3 = T_2(s) \end{bmatrix} = \begin{bmatrix} G_{11}(s) & G_{12}(s) & G_{13}(s) \\ G_{21}(s) & G_{22}(s) & G_{23}(s) \\ G_{31}(s) & G_{32}(s) & G_{33}(s) \end{bmatrix} \begin{bmatrix} U_1 = Qb(s) \\ U_2 = R(s) \\ U_3 = Q_2(s) \end{bmatrix} \quad (1)$$

Equations (2) to (10) present the transfer functions obtained, which consist of the input/output relations presented in (1). For the calculations, the deadtime of these functions were expressed using a simple first-order Taylor series approximation.

$$G_{11} = \frac{0.69}{112s+1} \quad (2)$$

$$G_{12} = \frac{3.12e^{-13s}}{203s+1} \quad (3)$$

$$G_{13} = \frac{0.56e^{-1s}}{145s+1} \quad (4)$$

$$G_{21} = \frac{-0.08e^{-26s}}{172s+1} \quad (5)$$

$$G_{22} = \frac{-0.04e^{-4s}}{364s+1} \quad (6)$$

$$G_{23} = \frac{-0.06e^{-17s}}{135s+1} \quad (7)$$

$$G_{31} = \frac{0.02e^{-1s}}{74s+1} \quad (8)$$

$$G_{32} = \frac{0.14e^{-2s}}{407s+1} \quad (9)$$

$$G_{33} = \frac{0.02}{85s+1} \quad (10)$$

Experimental tests were carried out, data were evaluated and with the process equations the existing interactions were verified by controlling the process with and without the proposed approach. Experiments were carried out in order to construct the relative gain array matrix (RGA) (Shinskey, 1996) for the 2 x 2 system (reboiler and distillate temperature control loops) and for the 3 x 3 systems (reboiler, second stage and distillate temperature control loops). In this case, the cited method was used to identify the degree of coupling among the proposed systems and not to define the control structure, which is the usual purpose. This evaluation is important since the intermediate column stages also influence the temperature profile and the process composition.

Equation (11) presents the matrix obtained for the 2 x 2 system and (12) the matrix for the 3 x 3 system.

$$\Lambda = \begin{bmatrix} 0.89 & 0.11 \\ 0.11 & 0.89 \end{bmatrix} \quad (11)$$

$$\Lambda = \begin{bmatrix} 0.77 & 6.80 & -6.57 \\ -4.92 & -0.07 & 6.03 \\ 5.50 & -5.69 & 1.54 \end{bmatrix} \quad (12)$$

Although the selection of the best control structure is not the main objective of this study, the 2 x 2 system is adequate as shown in (11), where the sum of the matrix columns and lines are 1 (one).

In (12) we can observe some values above one for seven of the nine possible combinations of control loops, which indicates strong interactions in these combinations. It is well known that the closer the element is to +1, the weaker the interaction between the loops, and the elements with high modulus values indicate strong interactions between the loops, or it could be that the system is sensitive to parameter changes (less robustness).

With this phase completed, the studies were followed by experimental tests using multivariable control algorithms. As mentioned above, the common technique of controller adjustment by decoupling was used. This tuning was defined since it provides good results (Waller, et al. 2003, Liu et al., 2006). It is important to note that the objective is to improve the control of a new column and the operation approach, and therefore the application of techniques used industrially is the aim.

In decoupled control, it is implicit that the design objective is to obtain a system that reduces the interaction between the loops through specific additional controllers, called decouplers. These are used to improve the performance of multivariable control systems through interaction compensation, though they are sensitive to changes in the process and require detailed process models, which are often difficult to obtain. These disadvantages often limit the use of multivariable controllers industrially. However, the static decouplers approach can be designed from the gains in the process in steady state, which are easy to obtain and can be adjusted in the field (Lee et al. 2005). Because of these

advantages, the static decouplers approach was used, based on the gain of each control loop.

Besides the objective of the implementation and study of advanced techniques using the distributed approach, these studies were carried out to evaluate whether or not it is possible to work with the hypothesis that the interactions caused by the tray can be eliminated when the reboiler interactions are reduced or eliminated.

Figures 4 and 5 show the reboiler and distillate temperature profiles for the strategies applied. In these experiments, the disturbance was applied by decreasing the feed temperature.

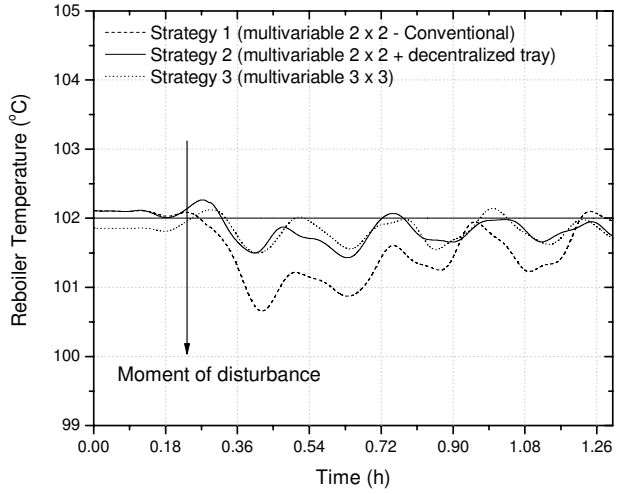


Fig. 4. Effect of the disturbance on reboiler temperature control loop response in relation to setpoint.

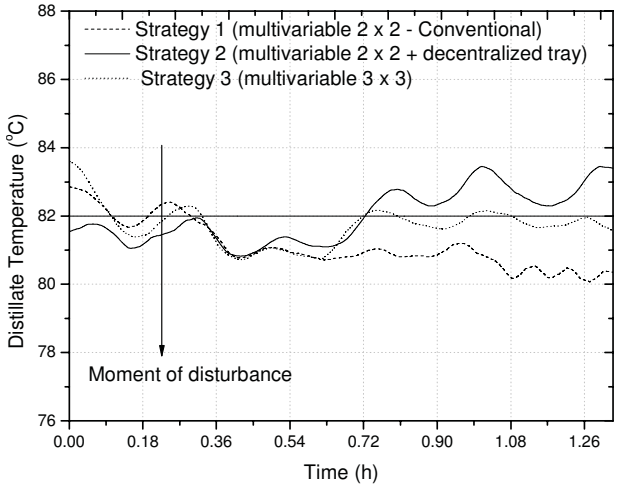


Fig. 5. Effect of the disturbance on distillate temperature control loop response in relation to setpoint.

It can be observed, in both figures, that the disturbance is quickly rejected when the distributed approach is used. Both strategies 2 (2 x 2 multivariable adjustment and decentralized adjustment at stage 2) and 3 (3 x 3 multivariable adjustment) showed that the steady state was reached faster than with strategy 1 (conventional multivariable control – 2 x 2 system). This result indicates that the distributed control action maintains the temperature profile in the column and thus it allows the reduction of the transients generated.

However, strategy 2, which assumes that the interactions between the tray temperature control loops and the other quality controllers is weak, leads to a value slightly higher than that desired. This case is better observed in relation to the distillate temperature. For this same variable, the disturbance applied was not completely rejected using the conventional control, and the distillate temperature stabilized at a lower value.

When reboiler and distillate temperature loops are evaluated together, the 3 x 3 distributed approach allows a better performance. It is possible that this adjustment, considering the interactions between the three control loops, made the system a bit slower, although it is still faster and less oscillatory than the conventional approach.

Figure 6 gives the second stage temperature profile, where the distributed control was implemented. As expected, the performance of the 2 x 2 multivariable adjustment strategy with decentralized adjustment at stage 2 leads to a value closer to the set point. In fact, using a decentralized PID controller at this stage, leads to faster dynamics than applying a multivariable 3 x 3 system which consider all interactions of this tray with the reboiler and distillate temperature control loops. However, strategy 3 showed a slight overshoot and rejected the disturbance quickly, in contrast to the conventional strategy.

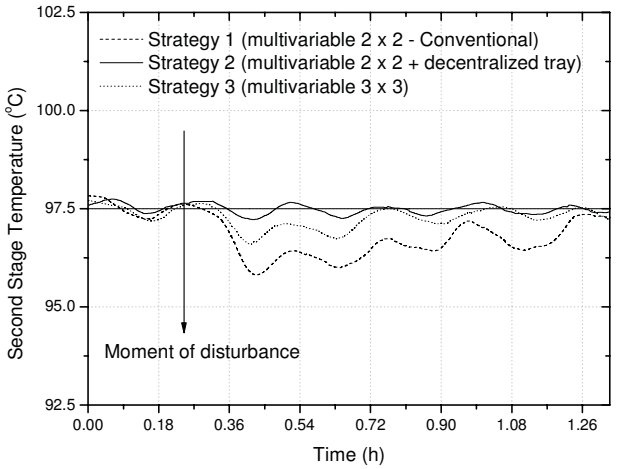


Fig. 6. Effect of the disturbance on the tray 2 temperature control loop response in relation to the setpoint.

In order to carry out a final evaluation regarding which strategy leads to the best performance, the effect of the disturbance on the temperature of the accumulator tank was studied. Since this is the last unit stage, it is the one with the highest transition time. Therefore, its behavior was observed by analyzing the temperature derivative in relation to the time required for the disturbance rejection, as illustrated in Figure 7.

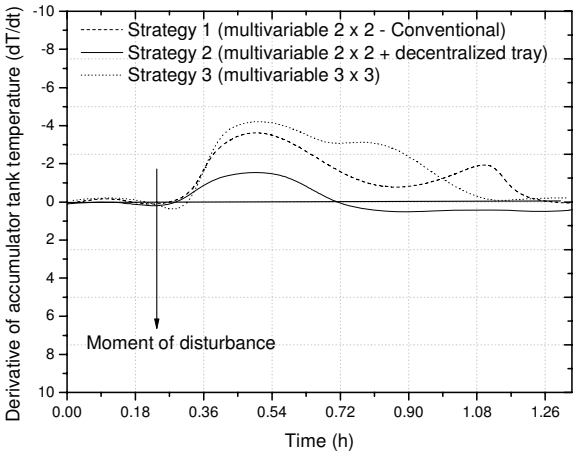


Fig. 7. Derivative of the temperature of the reflux accumulator tank in relation to the setpoint.

The figure demonstrates that the time required to reduce the effects on feed temperature disturbance is shorter when the distributed control approach is applied, considering the second stage temperature control loop not interacting with the others. This hypothesis will be true if the interaction at this stage can be eliminated by the reboiler temperature control loop decoupling. However, it is important to note that the value of the accumulator tank temperature did not return to the same steady state present before the disturbance. It is possible that the use of a PID decentralized controller at stage 2 allowed the production of a greater vapor phase inside the column as the temperature of the last stage was higher in this case. If this occurred, the condenser would produce more distillate and the accumulator temperature tank would stabilize at a different value, as was in fact observed.

4. CONCLUSIONS

The evaluation of the conventional and distributed approach, for a feed temperature disturbance, allowed a reduction in the column transition time and in the oscillations of the controlled variable when the strategy with control at stage 2 was used (independent of the tuning of this loop – decentralized or not).

It is also necessary to consider that the decentralized utilization of the temperature control loop which comprises the distributed approach gives better results than the conventional control system. This is an important result since most industrially implemented controllers are considered decentralized (Garelli et al. 2006).

The comparison between the use of a distributed control loop, decentralized or not with the reboiler and distillate temperature loops, shows that the hypothesis of weak interaction of an intermediate stage can be assumed. When a decoupler was used to tune the quality controllers of the base and top it is possible that the interactions with the trays were reduced. Thus, with the application of an advanced control algorithm, it is observed that the introduction of heat to one of the column stages allows a reduction in the operation time out of the desired conditions. As with classic controllers tested in previous research studies, the introduction of distributed heat throughout the column was shown to be a valid option for the reduction of transients, enabling faster dynamics and lower volumes of products processed out of the pre-defined quality parameters.

ACKNOWLEDGMENTS

The authors are grateful for the financial support of the National Agency of the Petroleum - ANP - and of the funding agency - FINEP - through the Program of Human Resources of ANP for the Petroleum and Gas Sector - PRH-34-ANP/MCT and to the National Council of Research - CNPq.

REFERENCES

Astrom, K.J.; Hägglund, T. (2001) The future of PID control. *Control Engineering Practice*, v. 9, p. 1163-1175.

Bezzo, F; Micheletti, F.; Muradore, R. Barolo, M. (2005) Using MPC to control middle-vessel continuous distillation columns. *Journal of Process Control.* v.15, p. 925-930.

Garelli, F., Mantz, R. J., Battista, H. De. (2006) Limiting interactions in decentralized control of MIMO processes. *Journal of Process Control.* v. 5, p. 473-483.

Jana, A. K.; Samantha, A. N.; Ganguly, S. (2005) Globally linearized control system design of a constrained multivariable distillation column. *Journal of Process Control.* v. 15, p., 169-181.

Karaca, S. (2003) Application of a non-linear long range predictive control to a packed distillation column. *Chemical Engineering and Processing*, v. 42, p. 943-953.

Koeijer, G.; Røsjorde, A.; Kjelstrup, S. (2005) Distribution of heat exchange in optimum diabatic distillation columns. *Energy*. v. 29, p. 2425-2440.

Lee, J., Dong, H. K.; Edgar, T. F. (2005) Static decouplers for control of multivariable processes. *AICHE Journal.* v. 51, p. 2712-2720.

Liu, T., Zang, W., Gao, F. (2006) Analytical decoupling control strategy using unit feedback control structure for MIMO processes with time delays. *Journal of Process Control***.** v. 17, p. 173-186.

Luyben, W.L. (2006) Evaluation of criteria for selecting temperature control trays in distillation columns. *Journal of Process Control* v. 16, p. 115-134.

Marangoni, C. Machado, R. A. F. (2007) Distillation tower with distributed control strategy: Feed temperature loads. *Chemical Engineering an. Technoly.* v 30, p 1292-1297.

Nooraii, A.; Romagnoli, J. A.; Figueroa, J. (1999) Process, identification, uncertainty characterization and robustness analysis of a pilot scale distillation column. *Journal of Process Control*, v. 9, n. 3, p. 247-264.

Shinskey, F. G. (1996) Process Control Systems. McGraw-Hill, 4a edition, New York, 439p.

Waller, M., Waller, J. B., Waller, K. V., (2003) Decoupling revisited. *Industrial & Engineering Chemical Research.* v. 42, p. 4575-4577.

Zhu, Y. & Liu, X. G. (2005) Dynamics and control of high purity heat integrated distillation columns. *Industrial & Engineering Chemical Research.* v. 44, p.8806-8814.

Simultaneous Synthesis, Design and Control of Processes Using Model Predictive Control

M. Francisco*, S. Revollar, **, P. Vega*, R. Lamanna**

**Departamento de Informática y Automática. Universidad de Salamanca. Spain (e-mail: mfs@usal.es).*
***Departamento de Procesos y Sistemas. Universidad Simón Bolívar. Venezuela (e-mail: srevolla@usb.ve)*

Abstract: This work presents the simultaneous synthesis, design and control of an activated sludge process using a Multivariable Model-based Predictive Controller (MPC). The process synthesis and design are carried out simultaneously with the MPC tuning to obtain the most economical plant which satisfies the controllability indices that measure the control performance (H∞ and l1 norms of different sensitivity functions of the system). The mathematical formulation results into a mixed-integer optimization problem with non-linear constraints that is solved using a real coded genetic algorithm. The solutions reflects the effect of applying different bounds over the controllability norms. The results are encouraging for the development of integrated design approaches with advanced control schemes which usually results in complex optimization problems difficult to solve with conventional techniques.

Keywords: Process Design, Controllability indices, Model Based Predictive Control, Genetic Algorithms.

1. INTRODUCTION

The fact that the incorporation of controllability issues at the early stage of process design improves the dynamical behaviour of the plants have motivated the development of different methodologies to deal with the simultaneous process and control system design as Kookos and Perkins (2001), Revollar et al. (2004), Sakizlis et al (2004), Francisco et al. (2005) and more recently Tlacuahuac-Flores and Biegler (2007) and Tlacuahuac-Flores and Biegler (2008).

The simultaneous process and control system design leads to a non linear optimization problem where economic objectives, operability specifications and control performance are considered. The most comprehensive applications contemplate, also, the process synthesis or the control structure selection resulting into a mixed-integer-non-linear optimization problem (MINLP). The controllability analysis might require the evaluation of dynamic performance indices, which translates the problem into a mixed-integer-dynamical optimization (MIDO).

Even thought, the contribution in the field of integrated design is considerable, most of the approaches use conventional PID controllers. Only few works (Sakizlis et al., 2003; Sakizlis et al., 2004; Francisco and Vega, 2006) have been addressed to the application, in the integrated design, of advanced control techniques as Model-Based Predictive Controllers (MPC). The reason is that the advanced control schemes involve solving an optimization problem on-line, leading to a drastic increase in the complexity of the design framework (Sakizlis, et al, 2003; Sakizlis, et al, 2004).

Model based predictive control (MPC or MBPC) makes use of a process model to calculate the optimal control law. The MPC have been mainly accepted due to its natural way of incorporating operating constraints in multivariable process and the successful results in industrial applications (Maciejowsky, 2002; Qin and Badgwell, 2003). The shortcomings of conventional control schemes can be overcome by pursuing an advanced model-based predictive control (MPC) scheme (Sakizlis et al, 2003).

There are several strategies to deal with automatic tuning of MPC based on optimization of dynamical performance index (Ali et al., 1993; Francisco et al, 2005; Li and Du, 2002) but its evaluation requires time-consuming dynamical simulations which is an important drawback of these methodologies. Vega et al (2007) proposed the use of frequency domain methods as controllability indexes to speed up the MPC automatic tuning procedure by solving a mixed sensitivity problem with constraints. It avoids dynamical simulations but the use of linearized models, caused some problems of stability and robustness in the presence of nonlinearities and load disturbances on the process.

The aim of this work is to perform the integrated synthesis and design of a process using model-based predictive controllers (MPC) that will be tuned automatically using the strategy proposed by Vega et al (2007). The activated sludge process of the Manresa's plant was selected to apply the integrated design methodology as has been done in previous work using a conventional PI control technique (Revollar, et al., 2004, Revollar, et al., 2005). Francisco and Vega (2006) applied advanced control techniques for the integrated design of the

mentioned plant, but the structure selection was not taken into account in the problem formulation.

The main difficulty for solving the problem is the existence of continuous process and MPC variables, integers for the prediction and control horizon and binary variables for the structural decisions, which leads to a complex mixed integer non linear optimization problem. Therefore, it is necessary the use of advanced algorithms that handle both, continuous and discrete decisions, to lead the optimization to economically optimal processes operating in an efficient dynamic mode around the nominal working point.

Several deterministic mathematical programming optimization techniques have been used for solving the simultaneous design and control problem (Sweiger and Floudas, 1997; Kookos and Perkins, 2001, Sakizlis et al, 2003; Sakizlis et al, 2004) but complex formulations and a considerable computational effort are required for its implementation. On the other hand, stochastic optimization methods as genetic algorithms have been a good alternative for solving such difficult problems with a minimum effort for its implementation. A genetic algorithm has been proposed for the solution of this non linear mixed integer optimization problem.

The paper is organized containing, first, the description of the MPC and the controllability metrics used for the automatic tuning in the integrated design procedure, the formulation of the optimization problem and the description of the process and in section 3. The analysis of the results is presented in section 4. Finally, conclusions and different projections of this work are included.

2. MPC FORMULATION AND CONTROLLABILITY METRICS

The basic MPC formulation consists of the on-line calculation of the future control actions by solving the following optimization problem subject to constraints on inputs, predicted outputs and changes in manipulated variables.

$$\min_{\Delta\hat{u}} V(k) = \sum_{i=Hw}^{Hp} \left\| \hat{y}(k+i\,|\,k) - r(k+i\,|\,k) \right\|_{Wy}^2 + \sum_{i=0}^{Hc-1} \left\| \Delta\hat{u}(k+i\,|\,k) \right\|_{Wu}^2 \qquad (1)$$

where k denotes the current sampling point, $\hat{y}(k+i\,|\,k)$ is the predicted output vector at time $k+i$, depending of measurements up to time k, $r(k+i\,|\,k)$ is the reference trajectory, $\Delta\hat{u}$ are the changes in the manipulated variables, *Hp* is the upper prediction horizon, *Hw* is the lower prediction horizon, *Hc* is the control horizon, *Wu* and *Wy* are positive definite matrices representing the weights of the change of control variables and the weights of the set-point tracking errors respectively. In this work the matrices *Wy* and *Wu* are diagonal but not time dependent, so the error vector $\hat{y}(k+i\,|\,k) - r(k+i\,|\,k)$ is penalized at every point in the prediction horizon and the changes in the control signal $\Delta\hat{u}(k+i\,|\,k)$ are penalized at every point in the control horizon.

The problem (1) is a Quadratic Programming (QP) problem that gives a sequence of control moves $\Delta\hat{u}(k+i\,|\,k)$. The first component of this sequence is applied to the system in time $k+1$, and the optimization problem (1) is repeated at the next sampling time (receding horizon strategy).

The MPC prediction model used in this paper is a linear discrete state space model of the plant obtained by linearizing the first-principles nonlinear model of the process Maciejowsky (2002):

$$\begin{cases} x(k+1) = Ax(k) + Bu(k) + B_d d(k) \\ y(k) = Cx(k) \end{cases} \qquad (2)$$

where *x(k)* is the state vector, *u(k)* is the input vector and *d(k)* the disturbance vector. Matrices ***A***, ***B***, ***Bd*** and ***C*** are of adequate dimensions. For this model the prediction is:

$$\hat{y}(k+i|k) = C\hat{x}(k+i|k) = C\left[A^i x(k) + \sum_{j=1}^{i} A^{j-1} Bu(k+i-j|k) \right] \qquad (3)$$

One reason for choosing state space models is that a significant part of the recent research literature on MPC shows contributions based on this type of models. Connections between the standard linear quadratic regulator (LQR) theory and unconstrained MPC when the horizons approach infinity could be another reason for that.

When the MPC controller is linear and unconstrained, it can be represented with a transfer function KMPC. The corresponding transfer function is:

$$u = \begin{pmatrix} K_1 & K_2 & K_3 \end{pmatrix} \cdot \begin{pmatrix} r \\ y \\ d \end{pmatrix} = K_1 r + K_2 y + K_3 d \qquad (4)$$

where Ki are the transfer functions between the control signal and the different inputs (r,y,d) which depend on the control system tuning parameters (Wu, Hp, Hw and Hc). Particularly, in our MPC formulation $K_2 = -K_1$ (Maciejowsky, 2002), then, control law can be stated as:

$$u = K_1(r - y) + K_3 d \qquad (5)$$

Consequently, taking into account control law and the transfer function of the open loop system, the closed loop response can be obtained from

$$y = \frac{GK_1}{1+GK_1} r + \frac{1}{1+GK_1} \tilde{d} \qquad (6)$$

where $\tilde{d}$ are the filtered disturbances

$$\tilde{d} = \left(GK_3 + G_d \right) d \qquad (7)$$

In order to state the automatic tuning problem, is necessary to define: The Sensitivity function *S(s)* between the load disturbances (*d*) and the outputs (*y*) and the Control Sensitivity transfer function *M(s)* between the load disturbances (*d*) and the control signals (*u*) when the reference is zero .

$$S(s)=\frac{y(s)}{d(s)}=\frac{k_3G+G_d}{1+GK_1} \tag{8}$$

$$M(s)=\frac{u(s)}{d(s)}=\frac{K_3-K_1G_d}{1+GK_1} \tag{9}$$

For solving the MPC optimization problem, MPC Toolbox of MATLAB has been used, with some specific modifications (Maciejowsky, 2002) implementing an extended state space representation.

Regarding to the controllability indices, some norm based metrics were considered. The first controllability index considered in this work is:

$$\|N\|_\infty=\max_w|N(jw)| \tag{10}$$

where N is a mixed sensitivity index that takes into account both disturbance rejection and control effort objectives. The function N is defined as:

$$N=\begin{pmatrix} Wp\cdot S \\ Wesf\cdot s\cdot M \end{pmatrix} \tag{11}$$

Wesf(s) is chosen to penalize control efforts adequately, and *Wp(s)* is chosen based on the spectra of disturbances to ensure proper disturbance rejection. *Wp(s)* and *Wesf (s)* are suitable weights for optimization. The selection of *Wp(s)* is explained below, and the weight *Wesf(s)* is selected to complete the H∞ mixed sensitivity problem and allows for the significance of control efforts. Note that control efforts rather than magnitudes of control are included in the objective function by considering the derivative of the transfer function *M(s)*.

In order to ensure disturbance rejection we need (considering normalized disturbances):

$$|S(jw)|\cdot|d(w)|<1 \tag{12}$$

in the disturbances frequency range where *S(jw)* is the frequency response of the sensitivity function, and *d(w)* is the disturbance spectra. By choosing a weight *Wp(s)* satisfying

$$20\cdot\log|Wp(jw)|^{-1}<20\cdot\log|d(w)|^{-1} \tag{13}$$

disturbance rejection can be assured imposing the following constraint in the optimization tuning procedure:

$$\|Wp\cdot S\|_\infty<1 \tag{14}$$

A typical choice for the weight *Wp(s)* is a rational function with one zero and one pole. *B* is the weight gain for high frequencies, *a* is the gain for low frequencies and w_b represents the required bandwidth for the closed loop system. The parameter *a* is very small to impose integral action to the system but avoiding numerical problems.

$$Wp(s)=\frac{\frac{s}{B}+w_b}{s+w_ba} \tag{15}$$

The maximum value of the manipulated variables (for the worst case of disturbances) can be constrained to be less than u_{max}, by means of the *l1* norm and the following condition:

$$\|M\|_1<u_{\max} \tag{16}$$

3. PROCESS DESCRIPTION AND PROBLEM FORMULATION

The activated sludge process was selected to study the simultaneous synthesis and control system design methodology. A simple model (Moreno et al., 1992) was selected, to avoid the excessive complexity of models like the ASM1 developed by the IAWPRC.

Moreno et al. (1992) model is based on the wastewater treatment process of the Manresa plant (Spain). It is founded in the classical Monod and Maynard-Smith model. It is assumed that the reactions take place in only one perfectly-mixed tank. However, in this work two possible structural alternatives consisting in one or two aeration tanks are considered.

The activated sludge process corresponds to the secondary wastewater treatment stage. In the aeration tanks or bioreactors, the activity of a mixture of microorganisms is used to reduce the substrate concentration in the water. The dissolved oxygen required is provided by a set of aeration turbines. Water coming out of each reactor goes to the settler, where the clean water is separated from the activated sludge that is recycled to both bioreactors. The control of this process aims to keep the substrate at the output (s_1 or s_2) below a legal value despite the large variations on the incoming substrate concentration (*si*) using the recycling flows qr_1 and qr_2 as manipulated variables (Moreno et al, 2002). The frequency and magnitude of the disturbances at the *si* input make the control of the plant a difficult task. The set of disturbances used for evaluate the control performance while tuning the MPC has been determined by COST 624 program Copp (2002).

3.1. Mathematical Optimization Problem

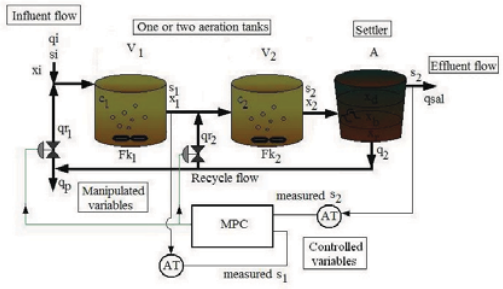


Fig. 1. Activated sludge process superstructure

The simultaneous synthesis, design and control of the activated sludge process pretend to obtain the most economical plant that satisfies the desired control performance. A cost function is defined to measure the economical issues while a predictive controller is tuned to achieve the desired closed loop behaviour according to the controllability norms described in section 2.

The two possible structural alternatives proposed for the plant are represented in a superstructure shown in figure 1. The model equations take the appropriated values for each structural alternative according to the binary y_1.

The mathematical formulation results into a mixed-integer-non-linear optimization problem where the objective is to minimize a cost function considering as decision variables: the structure (y_1), dimensions and controller parameters. Some constraints based in process model are set to find dimensions and initial working point, together with constraints over the norms used to measure the controllability of the plant with the actual controller parameters.

The cost function is:

$$f = p_1 \cdot (v_1 + v_2)^2 + p_2 \cdot A^2 + p_3 \cdot Fk_1^2 + p_3 \cdot Fk_2^2 + p_4 \cdot q_2^2 \qquad (11)$$

where v_1, v_2 are the reactor volumes and A is the cross-sectional area of the settler, Fk_1 and Fk_2 are the aeration factors for each reactor and q_2 is the overall recycle flow. The first three terms are associated to the construction cost that is proportional to the volume of the reactors and the area of the settler. The terms proportional to Fk_1, Fk_2 represent the aeration turbines costs, and the term proportional to q_2 represents pumping costs (purge and recycling).

Logical conditions must be imposed to guarantee the mathematical coherence of the model for any possible structure: if the second reactor does not exist: $y_1=0 \Rightarrow v_2=0$, $x_1=x_2$, $s_1=s_2$, $c_1=c_2$, $Fk_2=0$, $qr_2=0$, if the second reactor exist, then, $y_1=1$ and all the variables take values within their ranges.

The constraints imposed over mass balances in aeration tanks and the settler, are used to define the plant dimensions and the initial stationary working point.

$$\left| v_1 \frac{dx_1}{dt} \right| = \left| \mu_{\max} Y \frac{s_1 x_1}{(K_s + s_1)} v_1 - K_d \frac{x_1^2}{s_1} v_1 - K_c x_1 v_1 + q_{12} (xir_1 - x_1) \right| \le \varepsilon \qquad (17)$$

$$\left| v_1 \frac{ds_1}{dt} \right| = \left| -\mu_{\max} \frac{s_1 x_1}{(K_s + s_1)} v_1 + f_{kd} K_d \frac{x_1^2}{s_1} v_1 + f_{kd} K_c x_1 v_1 + q_{12} (sir_1 - s_1) \right| \le \varepsilon \qquad (18)$$

$$\left| v_1 \frac{dc_1}{dt} \right| = \left| K_{la} Fk_1 (c_s - c_1)\, v_1 - K_{01} \mu_{\max} \frac{s_1 x_1}{(K_s + s_1)} v_1 - q_{12} c_1 \right| \le \varepsilon \qquad (19)$$

$$\left| v_2 \frac{dx_2}{dt} \right| = \left| \mu_{\max} Y \frac{s_2 x_2}{(K_s + s_2)} v_2 - K_d \frac{x_2^2}{s_2} v_2 - K_c x_2 v_2 + q_{22} (xir_2 - x_2) \right| \le \varepsilon \qquad (20)$$

$$\left| v_2 \frac{ds_2}{dt} \right| = \left| -\mu_{\max} \frac{s_2 x_2}{(K_s + s_2)} v_2 + f_{kd} K_d \frac{x_2^2}{s_2} v_2 + f_{kd} K_c x_2 v_2 + q_{22} (sir_2 - s_2) \right| \le \varepsilon \qquad (21)$$

$$\left| v_2 \frac{dc_2}{dt} \right| = \left| K_{la} Fk_2 (c_s - c_2) v_2 - K_{01} \mu_{\max} \frac{s_2 x_2}{(K_s + s_2)} v_2 - q_{22} c_2 + W_1 \right| \le \varepsilon \qquad (22)$$

$$\left| AL_d \frac{dx_d}{dt} \right| = \left| q_{sal} x_b - q_{sal} x_d - A \cdot nnr \cdot x_d \exp(aar \cdot x_d) \right| \le \varepsilon \qquad (23)$$

$$\left| AL_b \frac{dx_b}{dt} \right| = \left| \begin{matrix} q_{22} x_2 - q_{22} x_b + A \cdot nnr \cdot x_d \exp(aar \cdot x_d) \\ -A \cdot nnr \cdot x_b \exp(aar \cdot x_b) \end{matrix} \right| \le \varepsilon \qquad (24)$$

$$\left| AL_r \frac{dx_r}{dt} \right| = \left| q_2 x_b - q_2 x_r + A \cdot nnr \cdot x_b \exp(aar \cdot x_b) \right| \le \varepsilon \qquad (25)$$

If the second reactor does not exist ($y_1=0$), the values of the variables given by the logical conditions mentioned above, annul the equations (20) and (21), a $W_1 = q_{22} \cdot c_2$ term is used to cancel equation (22).

The operation constraints for the activated sludge process are:

Residence times:

$$2.5 \le \frac{v_1}{q_{12}} \le 8 \qquad (26)$$

$$2.5 \le \frac{v_2 + (1 - y_1) \cdot W_2}{q_{22}} \le 6 \qquad (27)$$

where W_2 annul de constraint for $y_1=0$.

Mass loads in the aeration tanks:

$$0.001 \le \frac{q_i s_i + qr_1 s_2}{v_1 x_1} \le 0.12 \qquad (28)$$

$$0.001 \le \frac{q_{12} s_1 + qr_2 s_2 - (1 - y_1) W_3}{v_2 x_2} \le 0.12 \qquad (29)$$

where W_3 annul de constraint for $y_1=0$.

Sludge age in the settler:

$$2 \le \frac{v_1 x_1 + v_2 x_2 + AL_r x_r}{q_p x_r 24} \le 10 \qquad (30)$$

Limits in hydraulic capacity:

$$\frac{q_{22}}{A} \le 1.5 \qquad (31)$$

Limits in the relationship between the input, recycled and purge flow rates:

$$0.03 \le \frac{q_p}{q_2} \le 0.3 \qquad (32)$$

$$0.05 \le \frac{q_2}{q_i} \le 0.9 \qquad (33)$$

The controllability constraints are the limits over the norms described in section 2 where the transfer functions are referred to s_2 as the output, si y qi as the disturbances, and recycling flows qr_1, qr_2 as control variables. The parameter u_{max} is an upper bound for the magnitude of control variables. These constraints: $\|N\|_\infty < 1$, $\|Wp \cdot S\|_\infty < 1$, $\|M\|_1 < u_{\max}$ ensure a satisfactory control performance with the tuned MPC.

The main difficulties when solving this problem is the existence of continuous, integer and binary variables and the evaluation of controllability norms that implies the linearization of the process model for each possible solution. GA are particularly suitable, due to its robustness and the straightforward method to compute the objective function and constraints avoiding gradient evaluation.

4. RESULTS

For solving the problem using genetic algorithms (Gen and Cheng, 2000), a fixed length real coded chromosome is defined, containing the continuous normalized process variables, the controller parameters (Wu, Hp, Hc) and a binary variable to set the structure of the plant: [x_1, x_2, S_{sal}, xd xb xr qr_1 qr_2 qp Fk_1 Fk_2 v_1 v_2 A Wu Hp Hc y_1].

The location of the variables in the chromosome is important for the objective function and constraints evaluation procedure.

The genetic algorithm starts by generating randomly a population of possible solutions, that contains the same quantity of individuals for the two structural alternatives ($y_1=0$ and $y_1=1$). Each solution is manipulated to fulfil the logical conditions mentioned in section 3.1, according to the actual value of y_1. The new candidate solutions are manipulated also, according to the logical conditions. The population in the succeeding generation consists of 50% of the best individuals from the previous generation and 50% of the individuals generated by crossover.

The problem is solved using a population size of 200 individuals and 300 maximum iterations. Roulette selection and arithmetic crossover were used. The mutation rate decreases with generations from 0.1 to 0.02 and the crossover probability used is 85%. A penalization strategy is applied to deal with constraints. The genetic algorithm was run 10 times for each case study, giving optimal feasible solutions for each run with an average computing time of 9657 seconds.

Two scenarios with different demands on control performance were proposed. For the case 1 the norm $\|M\|_1 < 1000$ and for the case 2 the norm $\|M\|_1 < 1500$. The weights Wp for both cases are: $Wp(s) = \frac{8s+19.2}{s+0.0001}$

Table 1. Numerical results for integrated synthesis and design with MPC for the case 1

Cost (MU)	0.13	Wu	0.0122
V_1 (m^3)	8640	Hp	9
A (m^2)	2728.9	Hc	3
S_1 (mg/l)	115.66	$\|N\|_\infty$	0.97
Qr_1 (l/hr)	371.47	$\|M\|_1$	987.47
Fk1	0.035	$\|Wp \cdot S\|_\infty$	0.92
Residence times	2.5		
Mass loads	0.08		
Hydraulic capacity	0.55		
Sludge age	2.08		

The results for the two scenarios are presented in tables 1 and 2. In both cases, the transfer functions and weights are referred to disturbances in si and qi. It is observed that the solution gives small economical plants that satisfy all the process and control constraints. It is important to notice the flexibility of the method for different limits imposed over the constraints leading to plants of different dimensions. In the case 1, where an stringent bound is imposed over l1 norm is obtained a plant with 8640m^3 reactor while, for the case 2, with a relaxed bound in l1 norm is possible to obtain an smaller plant with a reactor of 5858.1 m^3 which is reflected in cost.

The optimization case 1 produces a plant with better disturbance rejection because the weight Wp_1 is more restrictive for sensitivity function S. On the other hand, with a smaller bound for $\|M\|_1$, the magnitude of control is less relaxed than in case 2 giving a smaller range of action to the manipulated variable to reject disturbances. The values of Wesf for qr_1 and qr_2 control sensitivity functions are fixed to:

$$Wesf_{qr1}(s) = \frac{0.0117s+0.14}{s+0.0004} \qquad Wesf_{qr2}(s) = \frac{0.0183s+0.22}{s+0.0004}$$

Table 2. Numerical results for integrated synthesis and design with MPC for the case 2

Cost (MU)	0.064	Wu	0.0069
V_1 (m^3)	5858.1	Hp	8
A (m^2)	2178.4	Hc	3
S_1 (mg/l)	118.06	$\|N\|_\infty$	0.979
Qr_1 (l/hr)	273.9	$\|M\|_1$	1454.9
Fk1	0.021	$\|Wp \cdot S\|_\infty$	0.786
Residence times	4.11		
Mass loads	0.0856		
Hydraulic capacity	0.65		
Sludge age	5.03		

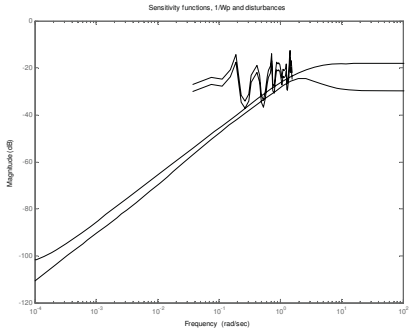


Fig.2. Sensitivity function S, Wp^{-1} and disturbances inverse spectrum for case 1

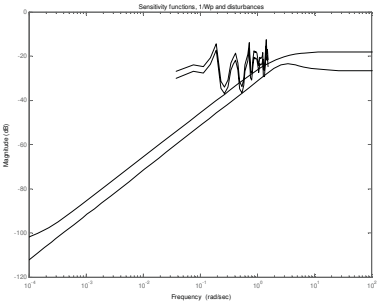


Fig.3. Sensitivity function S, Wp^{-1} and disturbances inverse spectrum for case 1

In figures 2 and 3 sensitivity functions S are presented for both cases. In the case 1 the inverse spectrum of disturbances is over Wp^{-1}, and in case 2 this weight is a bit more relaxed representing worse disturbance rejection. In figures 4 and 5 the dynamical responses of the optimal plants for both cases are

presented, to illustrate the better disturbance rejection for case 1 as have been previously mentioned.

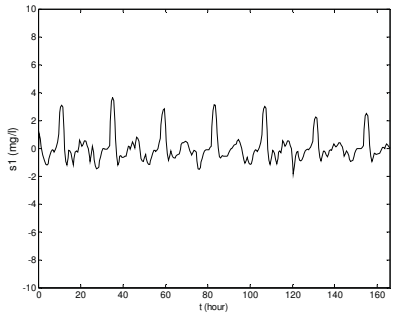


Fig. 4. Substrate response for case 1

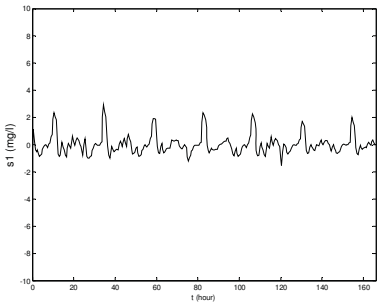


Fig. 5. Substrate response for case 2

5. CONCLUSIONS

In this work, the synthesis and integrated design of an activated sludge process with an advanced controller (MPC) was addressed. The problem was translated into a mixed-integer-non-linear optimization problem, with the evaluation of controllability norms to ensure the most economical design with a suitable control performance.

The MINLP was solved using a real-coded genetic algorithm which leads to good quality feasible solutions with desired disturbance rejection, which is the main control objective. The solutions obtained are sensible to the bounds imposed over controllability indices.

The controllability norms were set as constraints in the formulation of the optimization problem, but it could be formulated as a multiobjective optimization problem considering costs and controllability.

These results are encouraging for the development of simultaneous design and control approaches with advanced control schemes which usually results in complex optimization problems difficult to be solved. In this framework, the use of advanced control techniques represent a significant advance due to the advantages of these control strategies respect to conventional PID.

REFERENCES

Ali, E. and E. Zafiriou, 1993. On the tuning of Nonlinear Model Predictive Control Algorithms. Proceedings of the American Control Conference, pp. 786-790.

Copp, J.B., 2002, The COST Simulation Benchmark: Description and Simulator Manual. Office for Official Publications of the European Community.

Francisco, M., Vega, P. and O. Pérez, 2005. Process Integrated Design within a Model Predictive Control framework. IFAC'05 World Congress, Prague.

Francisco, M. and Vega, P., 2006, Diseño Integrado de procesos de depuración de aguas utilizando Control Predictivo Basado en Modelos. RIAI. Vol. 3, nº 4. 88-98.

Gen, M. and R. Chen, 2000, Genetic algorithms and engineering optimisation. J. W. and Sons.

Kookos I. and J. Perkins, 2001, An algorithm for simultaneous process design and control, Ind. Eng. Chem. Res. 40, 4079.

Li, S. and G. Du, 2002. On-line tuning scheme for generalized predictive control via simulation-optimization. IEEE International Conference on Fuzzy Systems, 1381-1386.

Maciejowsky, J. M. Predictive Control with Constraints. Prentice Hall, 2002.

Moreno, R. De Prada, C., Lafuente, J. Poch, M. and Montague, G. , 1992, Non-linear predictive control of dissolved oxygen in the activated sludge process. IFAC BIO 2pp. 289-298. Ed. Pergamon Press.

Qin, S. J. and Badgwell, 2003, T. A. A survey of industrial model predictive control technology. Control Engineering Practice11, 733-764

Revollar, S, Vega, P. and R. Lamanna, 2004, Algorithmic synthesis and integrated design of chemical reactor systems using genetic algorithms. Proceedings of World Automation Conference Vol 17, 493. Sevilla.

Revollar, S, Lamanna, R. and P Vega, 2005, Algorithmic synthesis and integrated design for activated sludge processes using genetic algorithms. ESCAPE. Barcelona

Sakizlis, S., Perkins, J., and E. Pistikopoulos, 2003. Parametric controllers in simultaneous process and control design optimization. Ind. Eng. Chem. Res., 42, 4545-4563.

Sakizlis, S., Perkins, J., and E. Pistikopoulos, 2004, Recent advances in optimization-based simultaneous process and control design. Computers and Chemical Engineering, 28, 2069-2086.

Schweiger, C and Floudas, C., 1997. Interaction of design and control: Optimization with dynamic models. In W. Hager and P. Pardalos (Eds.) Optimal control: Theory, algorithms and Applications. 388-435. Kluwer Academic Publishers.

Vega, P., Francisco, M., and E. Sanz, 2007. Norm based approach for automatic tuning of Model Predicitive Controllers. Proceedings of ECCE-6, Copenhagen.

Tlacuahuac-Flores, A and Biegler, L., 2008, Integrated Control and Process Design During Optimal Polymer Grade Transitions Operations. Computers and Chemical Engineering, In press.

An efficient multi-objective model predictive control framework of a PEM fuel cell

Ziogou Chrisovalantou[1], Simira Papadopoulou[1,3], Panos Seferlis[1,2] , Spyros Voutetakis[1]

[1]*Chemical Process Engineering Research Institute (C.P.E.R.I.), CEntre for Research and Technology Hellas (CE.R.T.H.), P.O. Box 60361, 57001 Thermi-Thessaloniki, Greece (Tel: +30-2310-498 317; e-mail:ziochr@cperi.certh.gr, paris@cperi.certh.gr)*

[2]*Department of Mechanical Engineering, Aristotle University of Thessaloniki, P.O. Box 484, 54124 Thessaloniki, Greece (Tel: +30-2310-498 169; e-mail:seferlis@cperi.certh.gr)*

[3]*Department of Automation, Alexander Technological Educational Institute of Thessaloniki, P.O. Box 14561, 54101 Thessaloniki, Greece (Tel: +30-2310-498 319; e-mail:shmira@teithe.gr)*

Abstract: Fuel cell systems can produce clean energy and have attracted the interest of both industrial and basic research in the recent years. They are part of a promising benign and environmentally friendly technology and they can be used both in mobile and stationary applications. A dynamic model was constructed and validated using experimental data based on a specific application, consisting of a high temperature PEM Fuel Cell (FC) working at a constant pressure and a Power Conversion Device that controls the current drawn from the FC. An integrated framework that consists of an online maximum power point prediction algorithm and a non-linear model based control scheme is presented. The proposed framework aims to maintain the fuel cell close to the optimum power point and the corresponding oxygen excess ratio level. Simulation studies show that the proposed control framework results in improved performance regarding the efficient and safe fuel cell operation under varying operating conditions.

Keywords: fuel cell control, model predictive control, power management

1. INTRODUCTION

Fuel cells are electrochemical devices that convert the chemical energy of a fuel directly into electricity and are under intensive development by several manufacturers. They are categorized according to the type of electrolyte used, operating conditions or fuel. The Polymer Electrolyte Membrane or Proton Exchange Membrane fuel cells (PEMFC) are currently considered by many to be in a relative more developed stage for ground vehicle applications and portable devices. PEMFC's have high power density, solid electrolyte, long cell and stack life, as well as low corrosion. Lately PEM fuel cells working at higher temperatures of up to 200°C have appeared such as those that use phosphoric acid doped polybenzimidazole (PBI) membrane, which is considered one of the most successful membrane systems so far [Jensen, 2007]. The benefits of operation at elevated temperature are mainly the tolerance to carbon monoxide concentration at the hydrogen feed, commonly present when operating with reformate streams, that can be increased by many orders of magnitude compared to that of a common PEM and also the water management which is a lesser issue, since the water is in vapor state. Additionally, the PBI membranes are conductive at very low relative humidity and consequently no moisture management is needed. Moreover, the high working temperature eliminates the possibility of water condensation in pores or channels of the fuel cell. Due to the higher temperature difference to the surroundings thermal management can be satisfactorily performed by a smaller cooling system [Jensen, 2007]. However, due to material limitations, the power of the fuel cell cannot be arbitrary used without prior consideration on the internal effects such as the provision for fuel and oxidant supply, temperature gradients, condition of the membrane (humidity) and so forth. The choice of the operating region leads to different characteristics for the unit regarding its profitability, effectiveness and safety. The dynamic response of a fuel cell is affected when the power demand fluctuates or when the fuel cell does not operate at its optimal steady-state design point [Golbert, 2007]. An optimization algorithm is used to search off-line for the optimum excess oxygen ratio level and the corresponding near maximum power. The primary objective of this paper is to demonstrate that model-based predictive control (MPC) is a suitable approach for efficient and safe fuel cell operation. The paper is organized as follows: Section 2 gives an overview of the dynamic fuel cell mathematical model. In section 3, the model validation procedure is presented. Section 4 presents the model-based predictive control structure along with the conventional control that is present for the pressures of the anode and the cathode compartments of the FC. Section 5 discusses the maximum power targeting algorithm The simulated results of the proposed MPC framework are presented and discussed in section 6.

2. MODELING AND ANALYSIS

The application is consisting of a high temperature PEM Fuel Cell working at a constant pressure and a Power Conversion Device capable of controlling the current drawn from the FC. In order to define a model based control strategy it is important to have an accurate model that reflects the transient dynamics and fuel cell system behavior and in the same time fast in execution in order to be useful for a real-time application. The mathematical model equations that describe the operation of the fuel cell consists of the voltage-current characteristics and a relationship for the consumption of the reactants as a function of the current drawn from the fuel cell. The main purpose of the detailed model is to describe the dynamic behavior in a way that the fundamental operating parameters current and pressure are established as manipulated variables and temperature as disturbance and power and excess oxygen ratio as controlled variable.

2.1 General

The main components of a PEM fuel cell are three - an anode, typically featuring platinum-containing catalyst, a thin, solid polymeric layer which acts as electrolyte, and a cathode, also coated with platinum [Mann, 2000]. In the PEM fuel cell the only reaction that takes place is the production of water from hydrogen and oxygen. In order to accurately describe the fuel cell behavior the mass balance and the equations that affect the voltage calculation are analyzed in the following section. The development of the fuel cell model is based on some assumptions. The gases are ideal and uniformly distributed inside anode and cathode. The stack is fed with hydrogen and air. The temperature is constant and uniform for each experiment. The gas channels along the electrodes have a fixed volume with small lengths, so that it is only necessary to define one single pressure value in their interior.

2.2 Electrochemistry and Voltage Calculation

Typical characteristics of FC are normally given in the form of polarization curve, which is a plot of cell voltage versus cell current density. To determine the voltage-current relationship of the cell, the cell voltage has to be defined as the difference between an ideal, Nernst voltage and a number of voltage losses and it is described in the current section. The main losses are categorized as activation, ohmic and concentration losses. The activation losses are caused by the slowness of the reactions taking place on the surface of the electrodes. A portion of the voltage generated is lost in driving the chemical reaction that transfers the electrons to or from the electrodes. The activation losses are described by the Tafel equation, which can be calculated as [Mann, 2000]:

$$\Delta V_{act} = \xi_1 + \xi_2 T + \xi_3 T \ln(C_{O_2}) + \xi_4 T \ln(i) \quad (1)$$

where ξ(i = 1–4) are parametric coefficients for each cell model. The term C_{O_2} is the concentration of oxygen on the electrolyte membrane at the gas/liquid interface (mol/cm^3), which can be expressed as [Zhong, 2008]:

$$C_{O2} = \frac{p_{O2}}{5.08 \cdot 10^6 e^{\left(\frac{-498}{T}\right)}} \quad (2)$$

The ohmic losses are caused by the resistance to the flow of electrons through the material of the electrodes and the various interconnections, as well as by the resistance to the flow of protons through the electrolyte. The ohmic losses are given by:

$$\Delta V_{ohm} = R_{mem} \cdot i \quad (3)$$

The ohmic resistance is described by:

$$R_{mem} = \frac{r_m \cdot mem_{thick}}{A} \quad (4)$$

where r_m is membrane resistivity (Ωcm) to proton conductivity, mem_{thick} is the membrane thickness (cm) and A is the active cell area (cm^2). Membrane resistivity depends strongly on membrane humidity and temperature, and can be described by an empirical expression given by Mann et. al. [Mann, 2000]. Finally the mass transport or concentration losses result from the change in concentration of the reactants at the surface of the electrodes as the fuel is used [Larminie J., 2003]:

$$\Delta V_{conc} = me^{ni} \quad (5)$$

where m and n are constants that can be estimated to give better fit to measured results. Thus, the actual voltage will be less due to the aforementioned losses that occur because of the various electrochemical phenomena. The Nernst voltage or open circuit voltage falls as the current supplied by the stack increases. The reversible thermodynamic potential is calculated using the Nerst equation and can be expressed as:

$$E = E^0 + \frac{RT}{2F} \ln\left[\frac{P_{H2} P_{O2}^{\frac{1}{2}}}{P_{H2O}}\right] \quad (6)$$

where F is the Faraday's constant (C/kmol) and p_i are the partial pressures (atm) (with i=H_2, O_2, H_2O). The equation that combines the above irreversibilities expresses the actual cell voltage:

$$V_{cell} = E - V_{act} - V_{ohm} - V_{conc} \quad (7)$$

The above equation is able to predict the voltage output of PEM fuel cells of various configurations. Depending on the amount of current drawn the fuel cell produces the output voltage according to (7). The electric power delivered by the system equals the product of the stack voltage V_{cell} and the current drawn I:

$$P = I \cdot V_{cell} \quad (8)$$

2.3 Mass Balance Equations

The model equations consist of the standard material balance of each component. Every individual gas follows the ideal gas equation. Therefore mass is described through partial pressures of each gas in the material balances:

$$\frac{d}{dt} p_g = \frac{R \cdot T}{V} \left[q_g^{in} - q_g^{out} - q_g^r \right] \quad (9)$$

where R is the universal gas constant (J (kmol K)$^{-1}$)), T is the temperature (K), V is the anode or cathode volume (1). For each gas q_g^{in} is the input flow, q_g^{out} is the output flow and q_g^r is the consumption or production due to the reaction. The same expression is used for oxygen, hydrogen and the produced water by replacing the term g with the corresponding gas. The amount of hydrogen consumed due to reaction is calculated as:

$$q_{H2}^r = \frac{I}{2F} \quad (10)$$

and for the oxygen :

$$q_{O2}^r = \frac{1}{2} \frac{I}{2F} \quad (11)$$

while the water production can be described by :

$$q_{H2O}^r = -\frac{I}{2F} \quad (12)$$

The water production rate is the same as the hydrogen's reaction rate, since water is produces as hydrogen is consumed. The oxygen reaction rate is the half of that of the hydrogen due to the stoichiometry of the reaction. As the load draws current, the reactants become depleted in the fuel cell and partial pressure of oxygen and hydrogen drop accordingly. A common practice to protect the fuel cell from reactants starvation is to supply it with excessive amounts of hydrogen and oxygen.

2.4 Oxygen Excess Ratio

There are two phenomena that can deteriorate or even destroy the fuel cell, flooding and oxygen starvation. Flooding is related to temperature and humidity, which are assumed constant and stable in the developed model since it is a high temperature FC where flooding is rather avoidable. The second one, the oxygen starvation, when it occurs the operation of the FC must be stopped in order to prevent fuel cell malfunction. The lack of oxygen is a complicated phenomenon that occurs when oxygen falls below a critical level at any location within the cathode. This phenomenon entails a rapid decrease in cell voltage, which in severe cases can cause a hot spot, or even burn-through on the surface of a membrane. To prevent this catastrophic event, the system must either remove the current from the stack or trigger a shut-down procedure. For all these reasons in a PEM fuel cell it is considered important to control the amount of available oxygen in the cathode. The air flow needs to be controlled rapidly and efficiently to avoid oxygen starvation and extend the life of the stack [Pukrushpan, 2004]. Although the oxygen concentration is not homogenous throughout the cathode, the control can be achieved by defining a parameter that indicates the oxygen level status in the cathode, named excess oxygen ratio level λ_{O2}. The excess ratio level is an unmeasured but observable variable that can be expressed as the inlet flow q_{O2}^{in} to the rate of oxygen consumption q_{O2}^r :

$$\lambda_{O2} = \frac{q_{O2}^{in}}{q_{O2}^r} \quad (13)$$

As can be observed by (11) and (13) the oxygen excess ratio level depends on the current drawn from the cell. This relationship can cause an abrupt and momentarily drop of the λ_{O2}, while it is related to the fuel consumption High values of λ_{O2}, and thus higher partial oxygen pressure improves the overall power. Low values of λ_{O2} indicate low oxygen concentration that could lead to oxygen starvation. Moreover, the temperature within the fuel cell may rapidly increase when oxygen concentration is too low. Therefore, the oxygen should be replenished quickly as it is depleted in the cathode [Vahidi, 2006].

3. PARAMETER IDENTIFICATION

The dynamic process model described in the previous section is validated using experimental data from a high temperature PEMFC. Nonlinear regression techniques are used to estimate the model parameters. The selected estimated parameters are the following: the parametric coefficient in activation losses (ξ_1) and the parameters in concentration losses (m, n). The characteristic cell voltage and the applied current density were measured through an on-line supervisory control and data acquisition system. Experiments were performed at the single cell system at constant temperatures between 170°C to 200°C. The activation area of the cell is 25cm2. The estimated values of the parameters are presented in Table 1.

Table 1 Estimated parameters

Parameter	Estimated value
ξ_1	-1.771
m	7.04E-05 V
n	9.44 E-03 cm^2 mA

Fig 1 compare the model predictions with the experimental data for various operating temperature levels for the polarization curves Fig 1 reveal that model predictions are in good agreement with the experimental data. As can be observed in the experimental results, a temperature increase raises cell voltage and consequently the fuel cell power output.

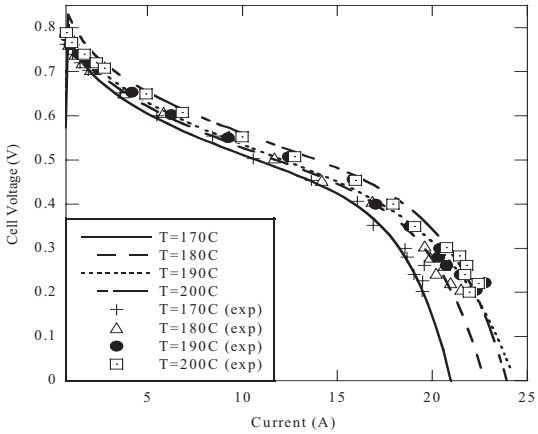


Fig. 1 Model predicted and experimental polarization curves

4. MAXIMUM POWER TARGETING

The power of the fuel cell depends nonlinearly on the applied current. As it is observed from the power curve there exists a unique operating point for each set of operating conditions, where the delivered power reaches a maximum power point (MPP). The operation of the system beyond MPP is not safe and should be avoided. The purpose of the control strategy is to deliver a near optimum power and at the same time to choose the proper operating region to ensure high fuel cell efficiency and avoid oxygen starvation. Thus a MPP tracking algorithm is developed that calculates the highest possible power as operating conditions vary. Fuel cell operation at the MPP is not very beneficial because the corresponding fuel efficiency is at best 50%. [Zhong, 2008]. As illustrated in Fig. 2 there exists an area where the power is near its MPP and the corresponding oxygen excess ratio level guarantees a safe and efficient fuel cell operation. The calculation of the MPP from process measurements is not possible as it depends on numerous factors that change during operation (e.g., relative humidity, gas mole fractions) and furthermore the entire power curve needs to be inferred to identify its maximum. Therefore, the fuel cell mathematical model is used in order to determine a desired trajectory towards the near MPP.

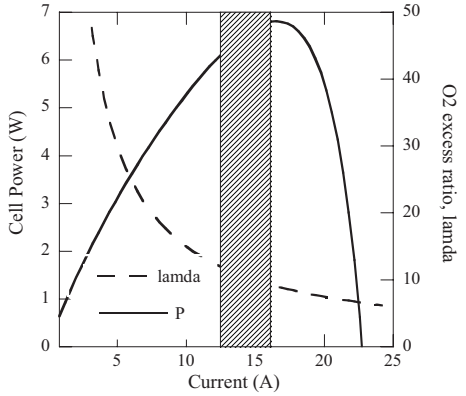


Fig. 2 Power curve and λ_{O2} (lamda) trajectory. Desired area of operation is shadowed.

Numerous methods have been proposed for maximum power tracking such as the perturbation and observe, adaptive extremum seeking algorithm, artificial intelligence methods and model-based methods to name a few [Zhong 2008, Krstic, 2000]. The current approach utilizes the developed non-linear dynamic model to determine off-line the near optimum region of operation and calculate the corresponding oxygen concentration level at the specific operating point usually defined as a function of temperature and pressure. The higher oxygen excess ratio level leads to a safer operation. The resulting strategy aims to an operation where a compromise between the maximum achievable power output and the optimal oxygen excess ratio is sought.

5. MODEL PREDICTIVE CONTROL FRAMEWORK

A Model Predictive Control (MPC) framework is formulated for the satisfaction of the control objectives described in the previous section. The fuel cell system presents a number of control challenges, the most significant of which is the nonlinearity in the area of the maximum power. Also an important control objective is the effective regulation of the oxygen concentration in the cathode. Furthermore, it is of interest to ensure safe operation during transients and sudden load changes. MPC is able to satisfy multiple control objectives under the presence of changes in process characteristics. Another important feature is its ability to deal with constraints. When a fuel cell operates near the MPP and consequently close to its operation limits constraints violations are critical in the achieved control performance.

5.1 Anode and Cathode Pressure Control

To regulate the anode and the cathode pressure a fast proportional-integral (PI) controller is implemented as used in the real system. The conventional PI controllers are used independently of the MPC scheme, which assumes that the anode and cathode pressure is held at a constant level as the dynamics of the PI control system are relatively fast. In the performed experiments it is assumed that the cathode and anode pressure is at 2 barg. These secondary loops are tightly controlled and assumed not to interact with the main control objectives of the system.

5.2 Model Predictive Control

Model predictive control (MPC) is part of a family of optimization-based control methods, which are based on on-line optimization of future control moves. Also MPC is based on the fact that past and present control actions affect the future response of the system. Using a process model, the optimizer predicts the effect of past inputs on future outputs. The deviation of the model prediction from the actual response is recorded and considered as the error of the process model, as shown in the block diagram of the MPC framework. The calculated error defines a bias term that is used to correct future predictions and it is constant for the entire prediction horizon. The block diagram describing the MPC scheme and the near optimum power targeting scheme is illustrated in Fig. 3.

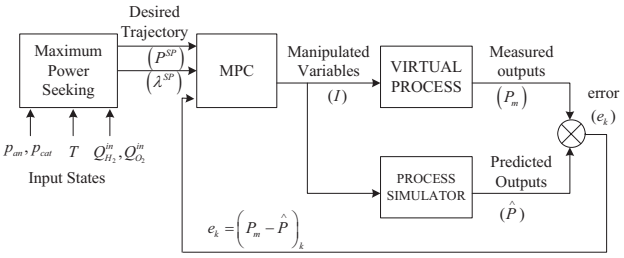


Fig. 3 Block diagram of the MPC control framework

The control structure utilizes two instances of the aforementioned dynamic model; the one corresponds to the Virtual Process (VP) and the second one to the Process Simulator (PS) or Model, and they are concurrently executed. Successive iterations between the optimizer, that evaluates the optimum value for the manipulated variable, and the model, that calculate the response of the process to the imposed control action are performed. The mathematical representation of the MPC algorithm is as follows:

$$\min_{k+j-1} J = \sum_{j=1}^{Np} \left\| \hat{P}_{k+j} - P_{k+j}^{SP} \right\|_{w_P}^2 + \left\| \hat{\lambda}_{k+j} - \lambda_{k+j}^{SP} \right\|_{w_\lambda}^2 \quad (14)$$

Subject to :

$$e_{k+j-1} = \left(y^{meas} - y^{pred} \right)_{k+j-1}, y_k = P_k \quad (15)$$

$$\hat{y}_{k+j} = y_{k+1}^{pred} + e_{k+j-1} \quad (16)$$

$$N_c = (T_c - T_k) / \Delta t_c \quad (17)$$

$$N_p = (T_p - T_k) / \Delta t_p \quad (18)$$

Where vectors $\hat{P}_k^{SP}$ and $\hat{\lambda}_k^{SP}$ denotes the desired response trajectories. The difference e_k between the measured variables y^{meas} and their predicted values y^{pred} at time instance *k* is assumed to persist constant for the entire number of time intervals N_p of the prediction time horizon T_p. While T_c denotes the control horizon reached through N_c time intervals. Also this minimization is subject to constraints on the manipulated and controlled variables:

$$I_{\min} \le I_{k+j-1} \le I_{\max} \quad (19)$$

$$\lambda_{O_2,\min} \le \lambda_{O_2} \le \lambda_{O_2,\max} \quad (20)$$

Eq (19) imposes a constraint to the input variables that corresponds to their physical limits. Eq (20) imposes a constraint on lamda to avoid starvation. Tuning parameters of the algorithm are the weight factors in the objective function (w_p, w_λ) and the length of the prediction and control horizon. The selection of the appropriate prediction horizon is mainly dictated by the time scale characteristics of the system. The computational time to reach a solution of the nonlinear dynamic program may affect the duration of the control interval.

6. SIMULATION RESULTS

The performance of the proposed MPC framework is evaluated through a number of simulated examples for a high temperature PEMFC. In all cases, unless otherwise stated, the system operates at constant temperature ($T_{sim} = T_{process} = 180^\circ C$) and constant cathode pressure ($p_{an} = p_{cat} = 2$barg). The influence of the controller tuning parameters on the closed-loop performance of the MPC is investigated. The main parameters are the control and the prediction horizons and the weighting factors of the power and oxygen concentration terms. Both prediction and control horizons were chosen equal to 15 seconds while the intervals of the control actions were chosen equal to 5 seconds. The length between two consecutive control actions (Δt_c) was selected according to the required computational time of the optimization problem.

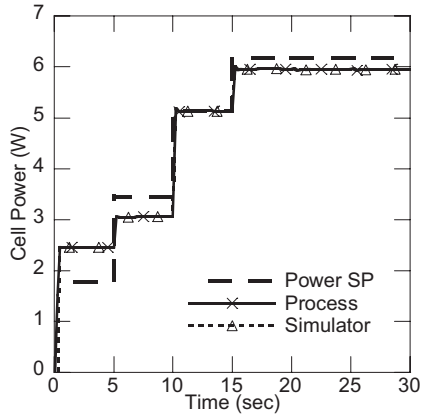


Fig. 4 Power response with unequal weights $w_p = 0.8,\ w_\lambda = 0.2$

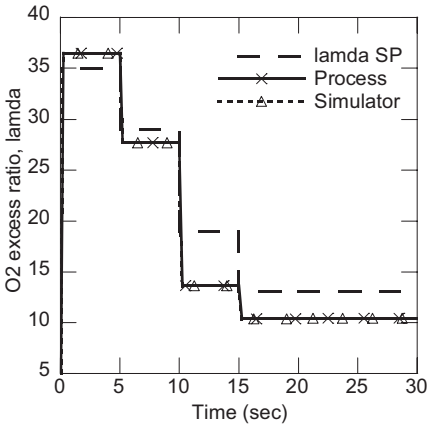


Fig. 5 Oxygen excess ratio level with unequal weights $w_p = 0.8,\ w_\lambda = 0.2$

Fig 4-7 show the sensitivity of the MPC performance on the weighting factors in the objective function. In the first case (Fig 4-5) more importance is given on the tracking of the power output while in the second case (Fig 6-7) an equal importance to both control objectives is imposed. In both

cases the desired setpoints were followed satisfactorily. However, in the first case excess O_2 is quite low which may cause difficulties in the fuel cell operation (i.e. oxygen starvation). The power output offset is small in the first case where a larger weight is used for the power output difference term. A better compromise is achieved in the second case with the excess oxygen closer to the desired level.

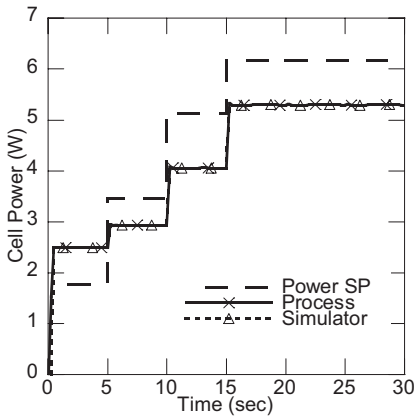


Fig. 6 Power response with equal weights $w_p = 0.5,\ w_\lambda = 0.5$

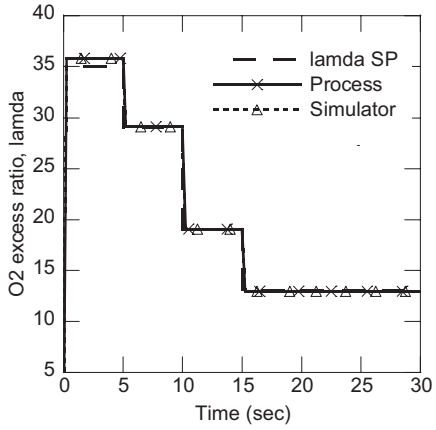


Fig. 7 Oxygen excess ratio level with equal weights $w_p = 0.5,\ w_\lambda = 0.5$

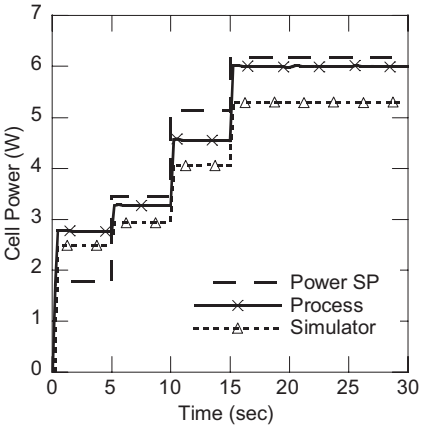


Fig. 8 Power response with altered process temperature

In another case study (Fig 8), significant mismatch in the fuel cell temperature between the Virtual Process and Process Model ($T_{sim} = 180^{\circ}C$, $T_{process} = 140^{\circ}C$) is deliberately introduced in order to asses the robustness of the proposed strategy to a significant disturbance. Fig. 8 illustrates the ability of the nonlinear MPC scheme to compensate for the temperature variation and successfully satisfy the control objective by close tracking of the desired power output level. The application of the constrained MPC framework allowed for an accurate targeting of the desired oxygen concentration and was able to give a near maximum power.

7. CONCLUSIONS

In this work a dynamic model for a high temperature PEM fuel cell stack based on single cell was developed and an advanced constrained predictive control framework was implemented. Having tested and verified some selected operational parameters a reliable MPC scheme was resulted. The MPC framework that combines two contradictive operational objectives, can safely lead to an operation that maximizes the power of a given size FC. The proposed MPC will be implemented and verified in the experimental fuel cell system. In order to improve the overall efficiency and safe operation, the controller would further include mathematical models for the auxiliary subsystems.

REFERENCES

Bordons Carlos, et. al (2006), Constrained Predictive Control Strategies for PEM fuel cells, *Proceedings of the 2006 ACC*, Minnesota, USA, June 14-16, 2486-2491

Jensen Jens Oluf, et al. (2007), High temperature PEMFC and the possible utilization of the excess heat for fuel processing, *International Journal of Hydrogen Energy*, 32 2007, 1567 – 1571

Golbert J., Lewin, D.R. (2007) Model-based control of fuel cells: optimal efficiency, *Journal of Power Sources,* 173 (1), 298-309

Krstic M., H.-H. Wang (2000), Stability of extremum seeking feedback for general nonlinear dynamic systems, *Automatica*, 36 (4) 2000, 595–601

Larminie J, Dicks A. (2003), *Fuel Cell Systems Explained*, 2nd Edition, John Wiley & Sons Ltd

Mann Ronald F., et. Al (2000), Development and application of a generalized steady-state electrochemical model for a PEM fuel cell, *Journal of Power Sources*, 86,173–180

Pukrushpan J., et. al (2004), Control of Fuel Cell Breathing, *IEEE Control Systems Magazine*, 0272-1708/04, 30-46

Vahidi A., et. al (2006), Current Management in a Hybrid Fuel Cell Power System: A Model-Predictive Control Approach, *IEEE Transactions on Control Systems,* vol. 14, no 6 2006, 1047-1057

Zhong Zhi-dan, et. al (2008), Adaptive maximum power point tracking control of fuel cell power plants, *Journal of Power Sources*, 176 2008, 259–269

Design of an Adaptive Self-Tuning Smith Predictor for a Time Varying Water Treatment Process

Khaled Gajam*, Zoubir Zouaoui,**
Philip Shaw*, Zheng Chen******

**Engineering Department, Glyndwr University, Mold Road, Wrexham, LL11 2AW, UK (e-mail: khaledgajam@hotmail.com)*
***Engineering Research Centre, Glyndwr University, Mold Road, Wrexham, LL11 2AW, UK (Tel. +44-1978-293151; e-mail: z.zouaoui@glyndwr.ac.uk)*
****United Utilities PLC, Huntington WTW, Chester Road, Huntington, Chester, CH3 6EA, UK (e-mail: philip.shaw@uuplc.co.uk)*
*****Engineering Research Centre, Glyndwr University, Mold Road, Wrexham, LL11 2AW, UK (e-mail: z.chen@glyndwr.ac.uk)*

Abstract: This paper presents the simulation and real time implementation of an adaptive predictive PI controller for the control of Chlorine dosing in secondary water disinfection rigs. This trial is part of a project that looks at the optimisation of process control specifically in the water industry. As one of the main treatment processes, Chlorine dosing is one of the processes that naturally imposes very long dead times for the controller to deal with. Although PI controllers are still commonly used for controlling this process, previous literature as well as trials carried as part of this project proved that the performance of PI controllers, no matter how tuned they are, is very sluggish and unreliable. A pilot rig was used instead of a live secondary disinfection rig, and a number of open loop step tests were performed for system identification. Once the process dynamics became known, complementary functions that estimate the process transfer function based on the water flow were introduced. A standard tuned PI controller configuration was simulated for the process, and then a Smith predictor was used in order to be able to compare the performance of the predictive PI with a standard PI controller. Tuning functions were derived for the PI to make it a self-tuning predictive controller, and parameter estimation functions were also used so that the final outcome is an adaptive self-tuning system. This system was then implemented on the same pilot rig, and real time implementation proved the findings obtained from the simulation. Both simulation and pilot rig tests show a very good dynamic response with excellent accuracy.

Keywords: Water process control; Chlorine dosing; PI tuning; Smith predictor; adaptive self-tuning.

1. INTRODUCTION

Process dead time is one of the most challenging issues in process control system design, as it makes the process difficult to control using standard feedback techniques mainly because the control action takes some time affect the controlled variable, and therefore the control action that is applied based on the actual error tries to correct a situation that originated some time before [1]. There can be several causes of this time delay, but in most process industries, the delay is caused by mass or energy transportation (also known as transport delay).

Processes with long dead times can not often be controlled effectively using a simple PI/PID controller. This is because the additional phase lag caused by the time delay tends to destabilise the closed loop system. The stability of such system can be improved by decreasing the controller gain, but that will certainly slow the controller down and make the response very sluggish [2]. Most water treatment processes incur relatively long transport delays, simply because the process variable is controlled by the addition of certain chemicals where a reaction time (Time taken for the chemical to dissolve/react in water) is to be allowed before the process variable can be sampled. It therefore, becomes part of the design requirements to allow enough distance between the chemical dosing and sampling points, and the time delay is then a function of the water flow within this distance as well as the flow in the sample line. A valid estimation of the delay may be expressed as:

$$Td = V/Q \quad (1)$$

Where Td is time delay, V is the volume of the pipe work between the dosing and sampling points, and Q is the water flow rate through the pipe. Another element that may have to be considered is the delay that might be caused by the sample line unless this delay is much smaller than the time delay caused by the pipe work. Previous research showed that PI controllers have been used to control dead-time processes, and the performance was acceptable when the dead time was small, but the performance deteriorates as the dead time increases, and in such cases a significant amount of detuning is required to maintain closed loop stability [5, 6].

2. THE CHLORINATION PROCESS

This research project focused on the implementation of predictive control for time varying water treatment processes that can generally be represented by a first order plus dead time (FOPDT) transfer function. Chlorine dosing is one of the essential treatment processes in most treatment plants as Chlorine is the most commonly used disinfectant and has been for the last hundred years [4]. The disinfection process takes place mainly in the main treatment plant, but there are usually secondary disinfection points spread around the distribution network in order to ensure that water provided to customers is microbiologically safe. Chlorine has many advantages that make it a more desirable disinfectant to use than other disinfectants (e.g. Ozone), but there are also disadvantages on using Chlorine. One of the main concerns in using Chlorine as a disinfectant is the formation of disinfection by products (DBP). Chlorine reacts with natural organic matter in water to form DBP's, and researchers are becoming increasingly concerned about the health problems those products can cause. A trade-off between the protection from infectious diseases and limiting exposure to DBP's has to be considered when using chlorine as a disinfectant [4]. This highlights the sensitivity of the Chlorination process, and the importance of having a reliable accurate control system to ensure that the amounts of Chlorine dosed in drinking water is what is needed.

3. PARAMETER ESTIMATION

The process described here is implemented on a pilot rig to study the effects of time varying processes on the performance of the controller. The advantage of using this pilot rig is the flexibility it gives in terms of controlling the parameters defining the dynamics of the process (i.e. water flow, pressure, and incoming Chlorine residual). These parameters have a direct impact on the process transfer function, and are normally hard to control in live disinfection rigs as they depend on water consumption by consumers.

Open loop step tests were performed at various flow rates and repeated several times. As expected, the variation of flow affected the dead time and the process time constant. Theoretically the dead time can be calculated using equation (1) provided that the precise volume of the pipe work as well as the precise flow rate are both known. Because of the relatively small size of the rig where even small inaccuracies in either the flow or the volume would affect the estimated process parameters, it was necessary to estimate the dead time using the open loop step test results. Figure 1 shows the relationship between the water flow rate and the process parameters.

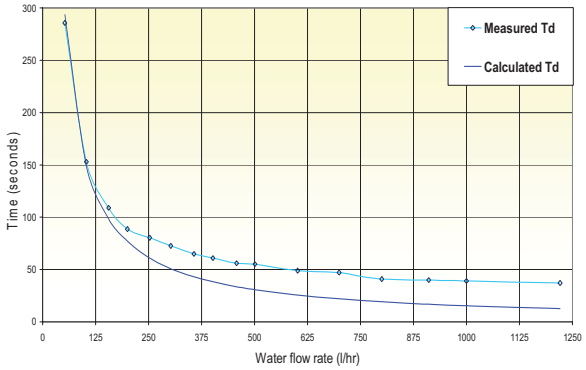


Fig. 1. Calculation and measurement of process dead time in relation to water flow rate.

It can be clearly noticed that at low flow rates, the measured dead time is almost identical to the calculated values, but as the flow increases beyond the lower 10% of this trial's flow range, the decrease in process dead time is not as expected. This highlights the importance of using real time data analysis when optimising the control system in this application.

4. PROCESS SIMULATION

Based on the open loop step test results, a model was built to represent both the process and the PI controller as implemented in most water treatment processes. Most chemical dosing process in water treatment use feedback plus feed forward control in a layout known in industry as Flow paced plus trim (Compound) control. The idea here is to combine the outputs of the feedback controller (usually a standard PI controller) and the feed forward controller (a gain element that is proportional to the water flow rate) in what will become the manipulated variable. In an ideal case, the feed forward controller output should keep the process variable equal to or very close to the set point. However, due to offsets in the dosing and measurement equipment, mismatch between the nominal and actual Chlorine strength, as well as the usual possibility of Chlorine presence in the incoming water, the feed forward controller will have to be supported by a feedback controller to act as a final trim and make up for all existing offsets.

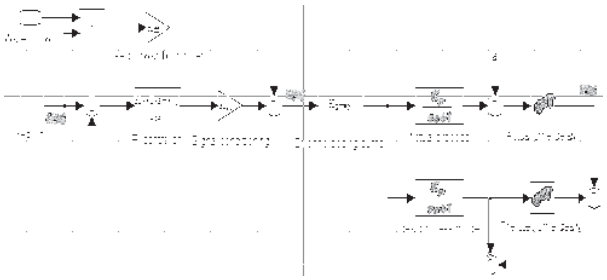


Fig. 2. Layout of Chlorine Dosing control loop.

A Smith predictor layout is shown in figure 2 combined with the PI controller and a proportional flow pacing function for feed forward control. The two main disturbances that can affect the process are the variation of water flow over time, and the incoming residual to the process. Many secondary disinfection rigs are installed on varying flow lines, where the flow rate changes over time by a ratio of up to 1:8. Therefore, it's important to consider the effect such big changes will have on the transfer function of the process, especially on the dead time of the process. Figure 3 shows a plot of real time data collected over a week from a time varying secondary dosing rig that is controlled by a PI controller.

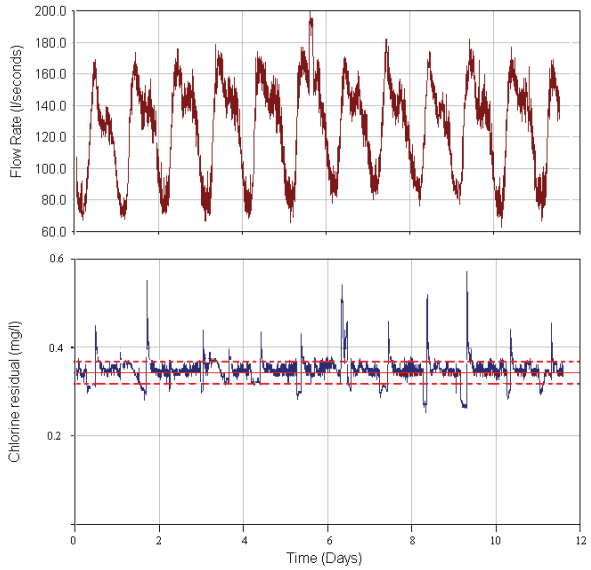


Fig. 3. Effect of time varying flow on the process variable (Chlorine residual).

The plot in figure 3 shows a clear link between changes in the water flow rate and variation in the Chlorine residual. Ideally, the Chlorine residual is supposed to remain within the band defined by the dashed lines for 99 % of the time for the system performance to be considered acceptable. Open loop step tests proved that the process time constant is also affected considerably by significant changes in the water flow rate. It is, therefore, obvious that in real applications, the practical approach would be to design a controller that will compensate for the inherent long dead times, and that will also be robust enough to the two major disturbances (i.e. Water flow rate changes, and incoming residual). For processes that are not time varying or where the variation is not that significant, PI controllers could produce what can be considered an acceptable performance. The more realistic scenario is the process where either one or sometimes all disturbances are significant, and that was the focus of this trial. The model in figure 2 was simulated. The process gain, time constant and dead time were all replaced with functions of the water flow rate that would make this an adaptive predictive controller. All the other gains are representative of the actual dosing and flow pacing constants. Figure 4 shows the closed loop step response of the simulated Smith predictor.

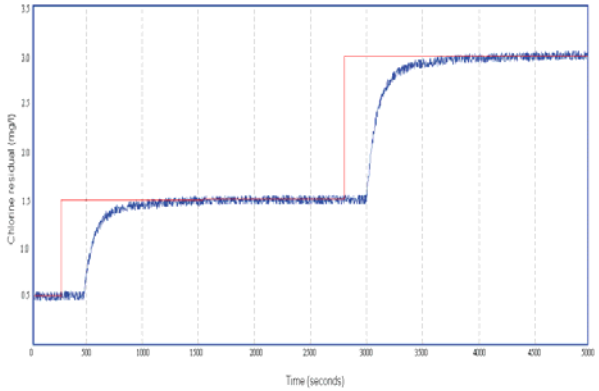


Fig. 4. Closed loop response of a simulated Smith predictor.

As mentioned earlier, the process parameters are flow dependent. A relatively low flow rate of 100 l/hr was chosen here to be able to simulate a challenging operational condition compared to a fast process. The above response shows that the controller performs well at set point tracking, and that accuracy and stability are maintained. The speed of response of the controller to the step changes is also very good considering the fact the dead time in this particular case was around 3 minutes. The same process model was simulated in closed loop using a standard PI controller that was also tuned using the same method that was used to tune the Smith predictor. It was noticed that at low flow rates, the dynamic response exhibits large un-damped oscillations as shown in figure 5.

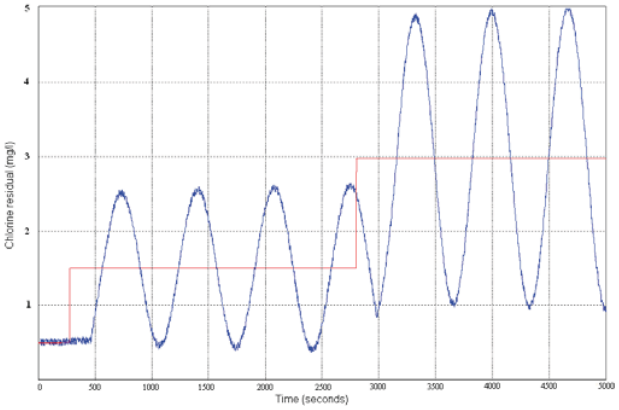


Fig. 5. Performance of a standard PI controller for a relatively long dead time process.

PI controllers are still being widely used in such processes. The only explanation as to why they are still being used is that in most cases the controller gain will be kept at a minimum value that is enough to allow the controller to track large process variable offsets, but at a very low speed of response. In a particular real time test on a live dosing rig, the response to a closed loop step test using a standard PI controller with a low gain took more than four hours to be completed due to the sluggish response of the controller.

5. IMPLEMENTATION AND TESTING

The water industry like most other process industries uses a combination of conventional analogue (4 – 20 mA) instruments and actuators with digital control PLC's, which nowadays have the resolution that allows them to control continuous processes. That said, PLC's still have their limitations when it comes to the implementation of advanced control methods as they seem to lack some important functionalities needed for continuous process control. It was also noticed with this particular application that offsets in the analogue modules interfacing between the PLC and the equipment (water flow meter, Chlorine analyser, and Chlorine dosing pump) can either exaggerate or dampen signals to and from the controller respectively. It is therefore important to obtain a process model that is a valid representation of the actual process so that the effect of the above mentioned offsets will be minimal; otherwise, those offsets will add to the uncertainties caused by the mismatch between the actual process and the process model and make the control system design process much more complicated.

The control algorithm was written in PLC ladder logic. Therefore, all continuous functions and the process model had to be written in that form. The dead time was then implemented using a data array that stores data in sequence at a frequency that is equal to the process sampling frequency, and the length of this array is equivalent to the predicted process dead time. To be able to test the Smith predictor for different flow rates, the control algorithm included functions to calculate the process parameters from online water flow rate measurements, and update the process model as well as the length of the delay array accordingly. There is also a tuning routine that uses the minimisation of the Integral of Absolute Error (IAE) tuning method to calculate the PI parameters (Kc and Ti). Many tuning methods have been suggested for Smith predictor applications, and many of them seem to have produced a robust performance [3, 5-8]. The following equations provide a good starting point for the minimum IAE method which is used to obtain the optimum PI parameters:

$$Kc = 0.984/Kp*(\tau/Td)^{0.986} \quad (2)$$

$$Ti = \tau/0.608*(Td/\tau)^{0.707} \quad (3)$$

Where Kc is the controller gain, Kp is the process gain, τ is the process time constant, Td is the process dead time, and Ti is the controller integration time. To make the controller self-tuning, these formulae were written in the control algorithm, and as part of the cyclic scan sequence, the PLC would calculate the process parameters based on the online flow rate measurement, then calculate the PI parameters as shown above and update them in the PI controller function block instantly. The following plot of real time data of closed loop step tests of the Smith predictor as explained above also shows the effect of sudden variation in the water flow rate on the process:

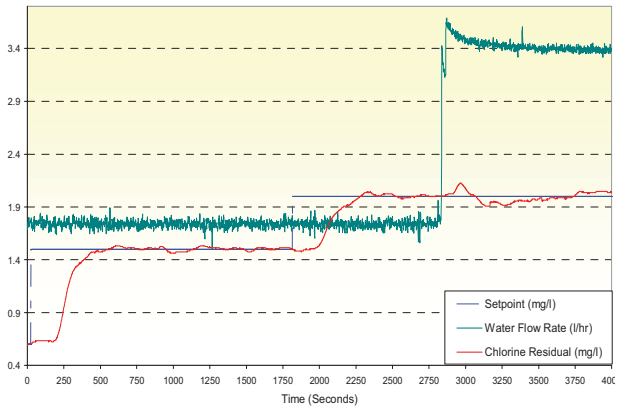


Fig. 6. Setpoint tracking performance of the Smith Predictor and its response to a major (Flow rate) disturbance.

The response achieved here conforms to the simulation results in terms of setpoint tracking, stability of the system, and the speed of its response. Also, the disturbance caused by the change in water flow rate did not have a significant effect on the process dynamics. The same test rig was used to test the same process under the same operating conditions using just a tuned PI controller in order to compare it to the performance of the Smith predictor, and the response is shown below.

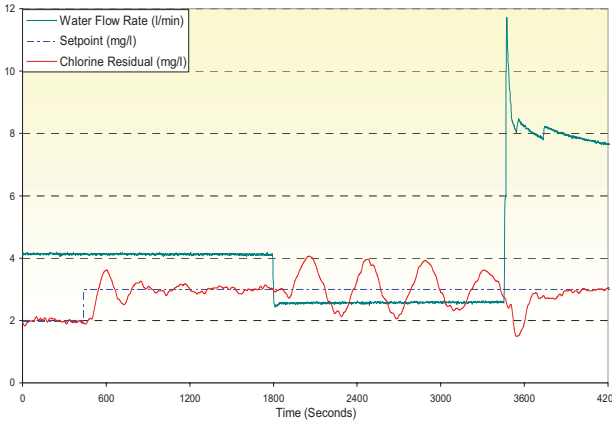


Fig. 7. Performance of a PI controller at various water flow rates.

The controller here was tuned well enough to produce an acceptable performance at a medium flow rate of 4.2 l/min, but when the flow was reduced to about 1.7 l/m, the controller became incapable of keeping the process variable at steady state until the flow was increased to a much high rate, where the controller slowly tracked the setpoint. Retuning the controller at this flow rate, would produce a faster response than the response shown, but at the low flow rate, the only option was to detune the controller (by keeping the controller gain as low as possible) so that it will respond to the step input, but at a relatively low speed of response, and hence not causing the oscillations seen in figure 7.

It is important to emphasise that without an accurate process model, the Smith predictor performance deteriorates. During this trial, initially the dead time was under estimated by around 20% and as a result the controller reacted earlier than required causing an overshoot in the Process variable of nearly 15%.

6. CONCLUSION

Simulation of PI controllers on FOPDT processes shows that the performance of the PI controller is dependent on the process dead time. In this trial the PI controller was found incapable of handling very long time delays regardless of the method used to tune it. The Smith predictor provides a reliable solution as long as the process model used is an accurate estimation of the actual process. In challenging process control applications where the process is time varying, adaptive Smith predictor configurations can be used effectively to overcome the uncertainty caused by the changes in process dynamics. Simulations and practical tests have shown that this method can be implemented successfully in water treatment processes.

REFERENCES

[1] Normey-Rico, J., & Camacho, E. (2008). Dead-time compensators: A survey. *Control Engineering Practice*, 16, 407-428.

[2] Kaya, I. (2002). A new Smith predictor and controller for control of processes with long dead time. *ISA Transactions*, 42, 101-110.

[3] Nortcliffe, A., & Love, J. (2003). Varying time delay Smith predictor process controller. *ISA Transactions*, 43, 61-71.

[4] Galal-Gorchev, H. (1996). Chlorine in water disinfection. *Pure & Appl. Chem.*, 68(9), 1731-1735.

[5] Guzman, J. L., Garcia, P., Hägglund, T., Dormido, S., Albertos, P., Berenguel, M (2007). Interactive tool for analysis of time-delay systems with dead-time compensators. *Control Engineering Practice*, 16, 824-835.

[6] Ingimundarson, A., & Hägglund, T. (2002). Performance comparison between PID and dead time compensating controllers. *Journal of Process Control*, 12, 887-895.

[7] Normey-Rico, J., & Camacho, E. (2008). Unified approach for robust dead-time compensator design. *Journal of Process Control*, 19(1), 38-47.

[8] Kaya, I. (2003). Obtaining controller parameters for a new PI-PD Smith predictor using autotuning. *Journal of Process Control*, 13, 465-472.

MODEL PREDICTIVE CONTROL OF A CRUDE DISTILLATION UNIT AN INDUSTRIAL APPLICATION

Serdar Kemaloğlu Emre Özgen Kuzu
Dila Gökçe Özgur Çetin

Turkish Petroleum Refineries Corporation, Izmit 41780
Turkey

Abstract: This paper reviews the application of a model predictive controller algorithm to a crude oil unit in Izmit Refinery of Turkish Petroleum Refineries Corporation as a summary of nearly eight months of practical study. The controller is designed to control the crude heating via process flows followed by furnace heating and distillation column with product strippers. After a process overview, the fundamental control loop considerations are discussed. Steps of determining inferred qualities and step test response tests are outlined. The controller design considerations, constraint handling and economic variables are presented. Finally, a comparison of before and after commissioning in the kerosene quality and rate is tabulated as well as simulation results of the system response for different set point changes in the product qualities.

Keywords: Advanced process control, multivariable predictive control, crude distillation processes, model identification.

1. INTRODUCTION

Model predictive control (MPC) strategies have found a wide range of applications from refineries to food processing and become a standard control algorithm for process industries. In 1999 nearly five thousand applications were reported with an increase of about 80 % in the following years (Qin and Bagdwell, 2003). MPC algorithm utilizes an explicit process model to optimize an open loop performance objective to constraints over a future time horizon, based on current measured variables and the current and the future inputs (Rossitier, 2003).

Turkish Petroleum Refineries Corporation (TUPRAS) is the largest industrial enterprise in Turkey. TUPRAS controls all of Turkey's refining capacity with operating four refineries with a total annual process capacity of 28.1 million tons crude oil. TUPRAS Izmit Refinery is the biggest refinery of four with annual refining capacity of 11.0 million tons of crude oil.

After privatization in 2006, implementation of advanced process control (APC) applications were initiated to increase operational excellence with many other projects. In this paper, an APC application to one of the crude units in Izmit Refinery is presented. After the process overview, base layer control strategies developed to sustain healthy layer for APC studies are explained. The main elements of the controller, controlled, manipulated and disturbance variables are discussed. Following the methods for obtaining inferred qualities, response tests and dynamic modelling strategies are outlined. The dynamic models obtained during tests are used in the commercial MPC algorithm SMOCPro. The control strategies, economic concerns and commissioning issues are discussed in the Section 4. The results of the commissioned plant data is tabulated as well as simulation of the controller for different set point changes in product qualities.

It is desired to give an outline of an industrial MPC application, but also some practical issues on implementation details. Finally, Section 5 outlines the study.

2. PROCESS OVERVIEW AND BASE LAYER CONTROL STRATEGIES

Plant-5 is one of the three crude oil units in TUPRAS Izmit Refinery and its simplified flow diagram is depicted in Figure1.

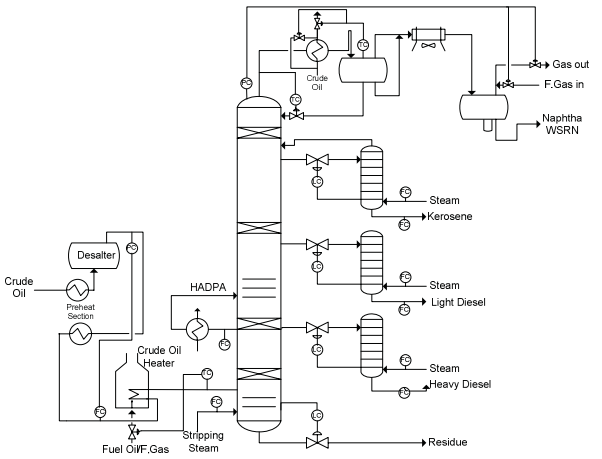


Fig. 1. CDU layout

The crude unit consists of mainly five operations. Advanced control strategies of steps 1,2 and 3 are in the scope of this paper:

1. Before and after desalter operation, the crude is heated via the hot streams available in the plant.
2. The crude is heated at a temperature around 320-350 ºC in two parallel furnaces.
3. The heated crude oil is distilled into products. Kerosene, light diesel and heavy diesel are drawn off after strippers. The bottom product, residue, is the feed to the vacuum unit.
4. The upstream of distillation column, a mixture of LPG, light straight run naphtha and heavy straight run naphtha, is fed to the naphtha splitter column where LPG and light naphtha are separated from heavy naphtha.
5. The liquid product of naphtha splitter upstream is separated into light naphtha and LPG in the debutanizer column.

Before implementation of advanced process control algorithm, a preliminary study comprising a base layer PID-based control is running successfully in normal operation. This study includes check of all instrumentation, checking for sticky valves, process feedback effects, process interactions, noisy measurements, retuning of PID controllers and possible configuration changes in the control strategies. In this scope, 56 regulatory PID controllers were retuned before starting testing.

Two base layer enhancements were carried out in distillation column control. There has been no bottom flow controller in the distillation column. Operators would manipulate the residue flow manually to keep level at operating ranges. This resulted continuous monitoring by operators and also discrete sudden changes in residue flow, which is the feed of vacuum unit as well. In the reviews, a new base layer of a level controller cascaded to the bottoms flow controller is implemented. With an appropriate tuning, the level is controlled while the bottoms flow does not vary significantly to create big disturbances to the vacuum unit.

The overhead drum temperature was manually controlled via bypassing from crude oil preheat heat exchangers at above 102 ºC to avoid corrosion problems. A new temperature controller has been implemented that closes the bypass valve in the line directing overheads product through the overheads cooler / crude preheat exchange.

3. CONTROLLER DESIGN

3.1 Variables

At the beginning of an APC project, the controller design requires selection of manipulated (MV), disturbance (DV) and controlled (CV) variables. These variables in this project:

- 28 manipulated variables are considered, 17 of which are for preheat and 11 for crude distillation column. These manipulated variables are feed flow controllers, furnace coil flow controllers and distillation column pressure, temperature and flow controllers.
- 3 disturbance variables are selected; amount of total feed and inlet flow rates to two furnaces.
- 19 controlled variables, where 4 variables are in preheat section and remaining in the crude distillation column.

Controlled variables for preheat section are desalter temperature and pressure and furnace outlet temperatures for the preheat section.

Some key variables are tabulated in Table 1 at the end of the paper. Distillation column controlled variables can be classified in three categories; inferred qualities, variables of economic importance and operational constraints.

The inferred qualities are the four main product qualities, heavy naphtha, light diesel and heavy diesel 95% distillation points and kerosene flash point. In order to control these qualities, inferential measurements have been developed. All the qualities have been inferred based on statistical regression of empirical data. The data was collected during a test period of two weeks, where manipulated variables and important column dynamics were changed one by one and held constant for two to three hours, while subsequent lab results were gathered. This data was used as a training period and inferential measurements were modelled using RQE Pro, software in the Process Control Technology Package (PCTP) of Shell Global Solutions International BV. Various combinations of process variables were regressed and good fits were achieved. Regular laboratory data of nearly eight months was also used to validate the model. The inferred qualities are mainly functions of pressure compensated temperatures. In some cases, vapour / liquid ratio below the draw-off tray of the product of which the quality was to be inferred. In the online application, the inferred qualities' equations' biases are updated with regular laboratory results. Before defining these qualities as a CV to the controller, several lab results were introduced to obtain a healthy estimation. These results correct any prediction error in the calculation that might arise in a local problem occurred during tests. Reducing the laboratory analysis amount after a certain period is another advantage of this method. Also in the online application, sudden change in bias may result in a change in the predicted quality, hence an aggressive MV output. To overcome this local disturbance, a filter time of thirty minutes to one hour has been applied to smoothen the update mechanism.

Operational constraints are usually the design specifications. Controlling column pressure valve opening prevents any loss of gas during operation. Reflux and heavy diesel pump around amount and top drum level as a CV sustains a safe operation. The level in the strippers is a major constraint where light diesel stripper level can be operated successfully to only a certain amount and is the main drawback of the light diesel amount.

Although they are not a part of the controller, reducing stripping steam amount and transferring the maximum heat from heavy diesel pump around are two economically beneficial CV's. The controller controls the stripping steam / feed ratio and heavy diesel inlet/outlet temperature difference multiplied by heavy diesel pump around amount at certain ranges.

3.2 Response Testing

During the commissioning of a predictive controller, testing and modelling efforts can take up to 90 % of the cost and time (Andersen and Kummel, 1992). Successful modelling engenders controller stability and performance on the predictive capability of the process model used. The most important element of a good modelling is the high quality clean data obtained form response testing. Also it should be noted that data analysis and model identification enhances process understanding and behaviour.

In the APC application in TUPRAS Izmit Refinery crude distillation unit, the response tests were carried out for three weeks in two shifts. For each of the 28 manipulated variables, tests were carried out separately in sequence, groups of six to eight steps were made in each. The sequence was repeated afterwards, obtaining a test data of sixteen to twenty moves for each variable. Repeating sequence allows preventing unmeasured disturbances that might happen in one of the sequences. The step sizes were defined in a way to see clear effects in the other variables, and steps were held for periods of one to one and a half of the settling time. The step sizes were changed sometimes to identify the presence of nonlinearities.

During step testing, it is very important to carefully observe the steps. The correlation of the independent variables, possible operator interventions and insufficient move sizes may result in poor data, hence poor modelling. Good signal-to-noise ratio enables that the effects of test changes in inputs could be clearly visible in process outputs. To overcome possible operator interventions, trainings were carried out before start of tests. Also changes in the regulatory configuration, PID tunings and possible controller saturations must be prevented during the test period.

3.3 Dynamic Modelling

Majority of industrial MPC applications use linear empirical models (Qin and Bagdwell, 2003). While analyzing the plant data, such an empirical dynamic process-modelling tool, AIDAPro –another PCTP software- was used. The results of step tests were analyzed and mathematically fit to obtain predictive process models.

In dynamic modelling, the interactions of all variables are taken into account, hence the effect of intermediate variables and disturbances can be tolerated in identification. Basically, multivariable higher order parametric models were created for each variable. These models are then approximated and reduced to a parametric model on individual relationship basis of two variables. Based on the data analyzed, the process knowledge and step test experience dynamic response of the variables are determined. Numerical representation of these dynamic response curves can be of any degree from first order zero gain to second order beta.

During modelling 31(28 manipulated and 3 disturbance variables) X 19 response curves were fitted. Of these 589 possible independent / dependent variable response curves, 53 responses are identified for the controller design. The other responses are either zero gain or insignificant.

4. CONTROLLER COMISSIONING

4.1 Controller Design

The dynamic models from observed plant data were used to design the controller. Shell Multivariable Optimizing Controller, SMOCPro is used to implement predictive controller in TUPRAS Izmit Refinery Crude Unit. SMOCPro, a part of the Process Control Technology Package (PCTP) of Shell Global Solutions International BV, and its algorithm was summarized by Qin and Bagdwell (2003) :

- An explicit disturbance model described the effect of unmeasured disturbances; the constant output disturbance was simply a special case,
- A Kalman filter was used to estimate the plant states and unmeasured disturbances from output measurements,
- A distinction was introduced between controlled variables appearing in the control objective and feedback variables that were used for state estimation
- Input and output constraints were enforced via a quadratic program formulation

As well as most advanced process control algorithms, SMOC algorithm also follows a reference trajectory by the future outputs on the prediction horizon and penalizes the control effort on the control horizon. General objective function of the controller can be written as

$$\min_{\Delta u(n)\ldots\Delta u(n+C-1)} \sum_{i=1}^{P} \left\| \hat{y}(n+i) - r(n+i) \right\|^2 w_1 + \sum_{j=1}^{C} \left\| \Delta u(n+i-1) \right\|^2 w_2 \quad (1)$$

in which 'u' represents inputs, 'y' is used to define outputs and the superscript ^ denotes the predicted values. Δu is the input variation and r is the reference trajectory of the outputs. In this optimization problem, the first term is used to minimize the error resulting from the difference between predicted outputs and reference trajectory during prediction horizon, P. The second term is the difference of control actions taken at each time step during control horizon, C. Weighting matrices w_1 and w_2 are positive definite matrices, with different magnitudes for all MV's and CV's. These matrices were used in controller tuning. The optimal input sequence's only first input is implemented to the system and the calculations are re-executed in the next sampling time.

The controller was tuned in offline program with simulations. While simulating both control considerations and economic variables, discussed in the next part, were considered.

Increasing w1, CV weights, increases the priority in decreasing the deviation in set point and reference trajectory, in other words makes the control tighter. Low w1 values allow bigger trade-offs. Increasing w2, MV weights, prioritizes minimum input variation and results in smoother moves.

As a base decision, the weights for column pressure, column top temperature and furnace outlet temperature controllers were given higher values than other MV's. The controller was expected to move these MV's smoother. CV weights for kerosene flash point, heavy diesel 95 % distillation and level of light diesel stripper were set higher than other weights to sustain tighter control. The weight tunings were completed by evaluating simulations.

4.2 Economic Variables

A key factor of SMOCPro is the economic function. Economic function is a bilinear function that the controller minimises. As long as the control objectives are met, the controller drives the defined economic function to minimum. Economic function in crude distillation unit consists of:

- Minimizing column top temperature, column pressure and stripping steam ratio to the feed
- Maximizing heavy diesel pump around duty, product draws to stripper level constraints and furnace heater duties. Maximizing the amount of heavy diesel by letting heavier cuts into heavy diesel and leaning to high limit. Maximizing the amount of kerosene by letting heavy naphtha into kerosene and approaching to low limit.

The constants of elements of economic function can be changed based on the plant needs or operational conditions.

4.3 Commissioning and Online Tuning

The controller was commissioned over a three- week period. The controller tunings determined in the offline studies were rechecked to prevent any model-process mismatch. After sustaining successful control, economic variables were commissioned by extending / restricting the limits. Operator trainings were also a major part of the commissioning period. All operators were trained to understand the basics of controller's function and philosophy.

4.4 Results

Over a two months period of controller running, the overall throughput of desired products increased significantly. Figure 2 shows the decrease in the naphtha yield, whole straight run naphtha (WSRN), and increase in the kerosene yield for a five weeks period of pre-commissioning and four weeks period of post-commissioning of the controller. The crude oil density was assumed to be constant, 32.46 and 32.43 API for pre and post-commissioning respectively. As seen from the figure, 11 % increase in the kerosene yield was achieved as a result of decrease in naphtha yield.

The change in the naphtha yield was also observed in the kerosene flash point. The naphtha in the kerosene product decrease the flash point of kerosene and by the high weight in the kerosene quality, the quality approaches to low limit. This approach shown in Figure 3 resulted in very significant economic benefits.

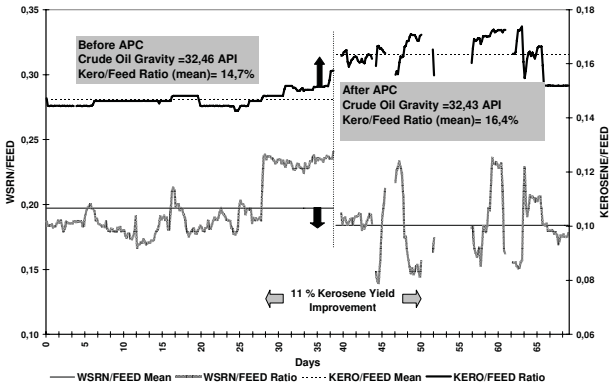


Fig. 2. Change in the naphtha yield before and after commissioning.

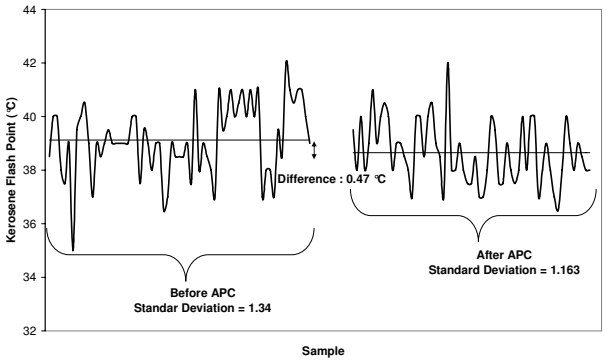


Fig. 3. Change in kerosene flash point before and after commissioning.

To illustrate the controller's performance on different situations, a simulation was studied using real plant data and specifications. When analyzing the results, a few points in controller tuning should be noted. Highest priority was assigned to kerosene and heavy diesel. Controller objective was to keep these qualities to low limit for kerosene and high limit for heavy diesel. On the other hand light diesel was assigned a low priority that its product rate was manipulated mainly in control of heavy diesel quality.

Figure 4 shows the changes in the product quality set points for kerosene flash point, heavy diesel, heavy naphtha and light diesel 95 % distillation points and their responses to these changes. The changes in critical column dynamics were reported in Figure 5. Figure 6 illustrates the product rate changes occurred during these set point changes.

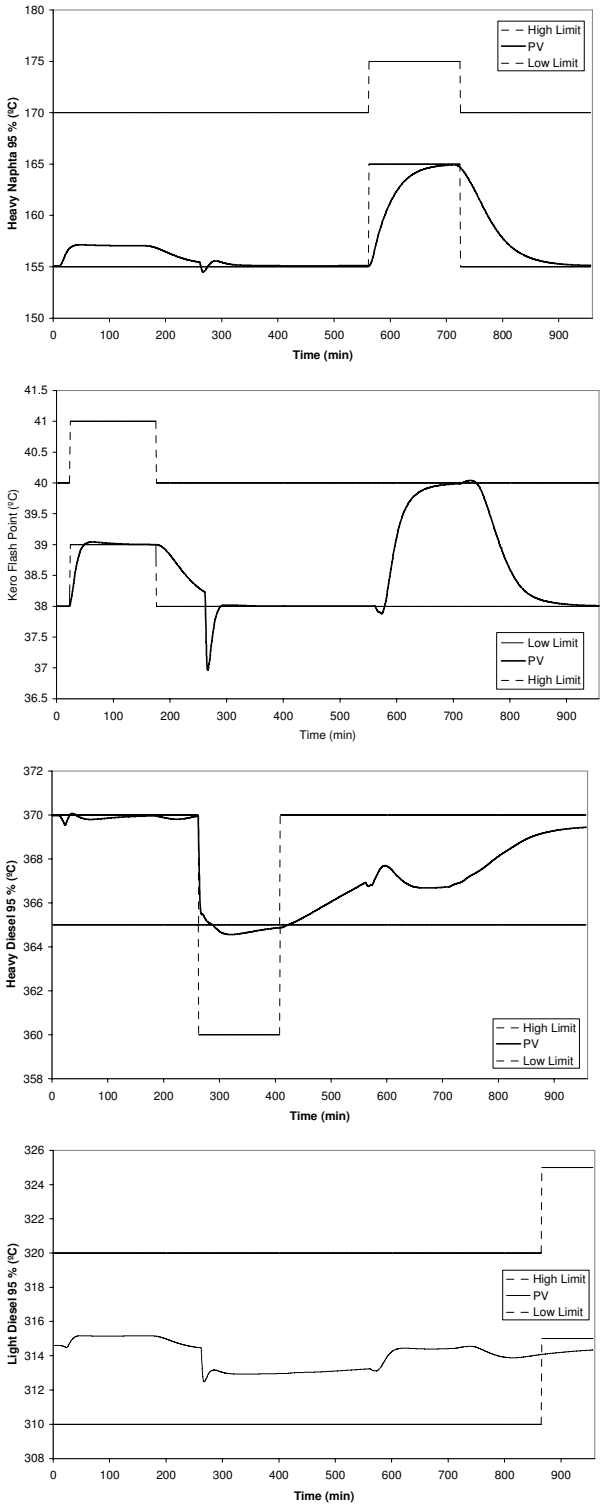


Fig. 4. Controller results in the qualities for the changes in the product quality limits.

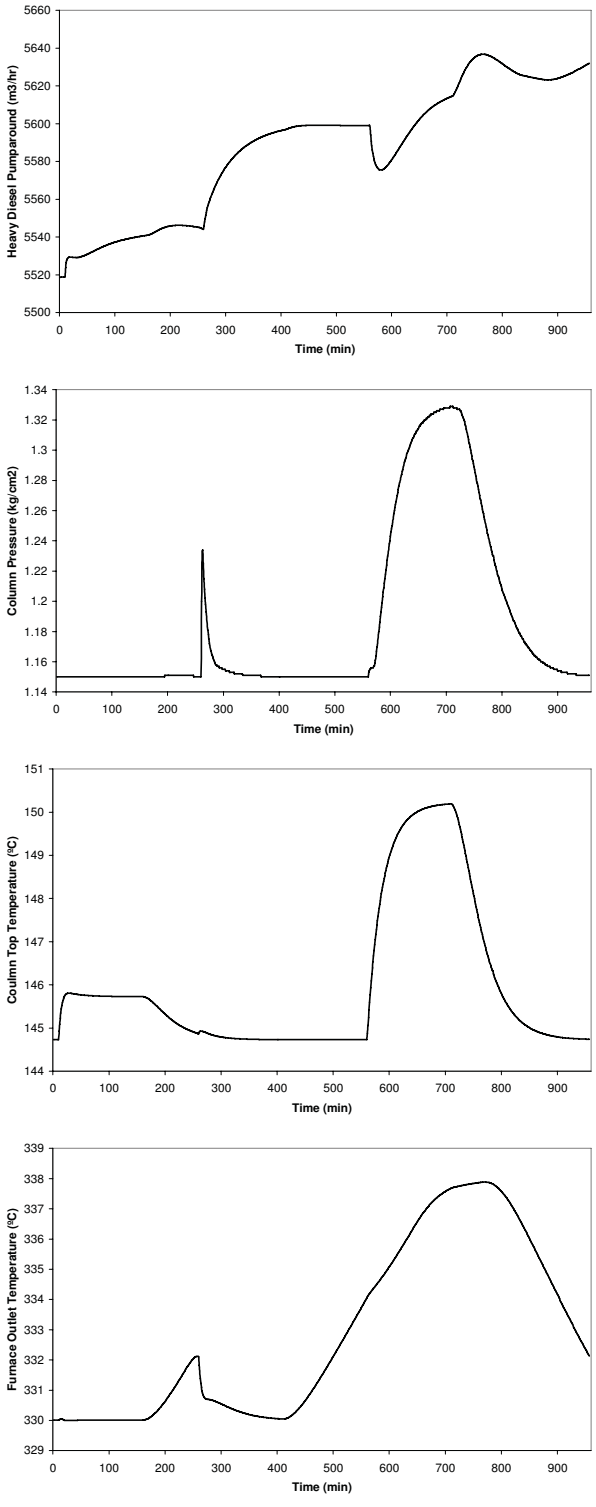


Fig. 5. Controller results in the distillation column critical controllers for the changes in the product quality limits.

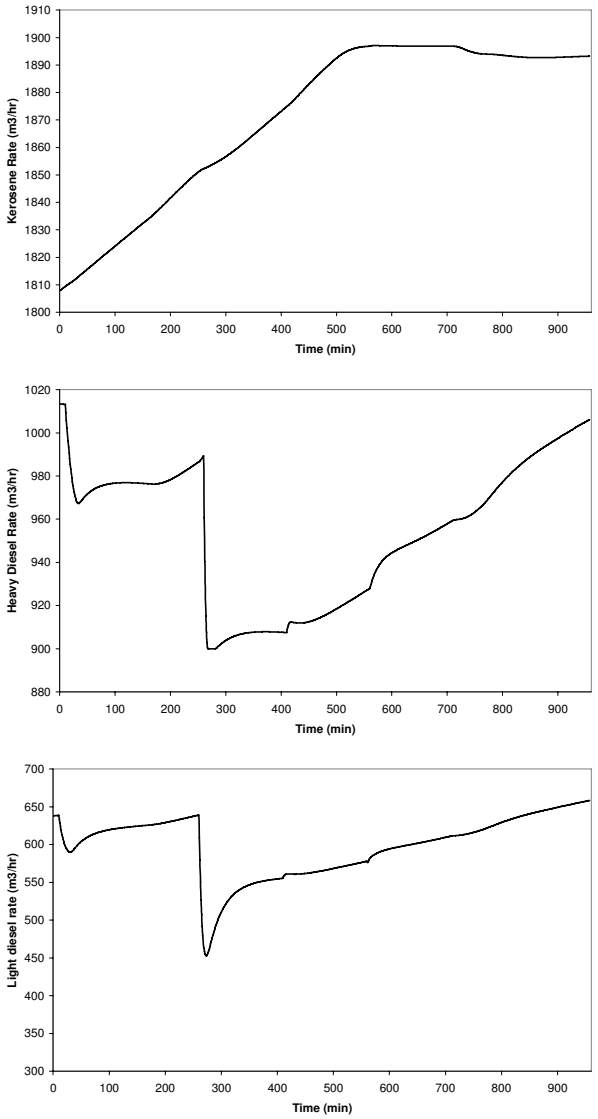


Fig. 6. Controller results in the product flow rates for the changes in the product quality limits.

5. CONCLUSIONS

In this paper, an industrial application of a model predictive controller algorithm has been presented. The controller has been designed for predictive control of crude oil preheat and distillation column of a crude oil unit in Izmit Refinery of Turkish Petroleum Refineries Corporation. A base layer control study has been performed to observe any nonconformity in the present control scheme. Tests have been carried out to obtain a mathematical model of the inferred product qualities. Dynamic step tests have been done for determined 28 manipulated variables. Dynamic modelling showed 53 responses to validate the controller.

Priorities for controlled variables and desired variation limits of manipulated variables were determined and the controller performance was tested using simulations in SMOCPro. Obtained model was commissioned for a three week period. It has been observed that the product throughputs and its economic benefit have been increased significantly.

Table 1. Control Variables of CDU Atmospheric Column

	Loop Description	MV	CV	DV
1	Heater Outlet Temperature	√		
2	Heavy Diesel Pump Around (HADPA) Duty		√	
3	Atm. Column Top Temperature	√		
4	Atm. Column Top Pressure	√		
5	Kerosene Draw-off Flow	√		
6	Light Diesel Draw-off Flow	√		
7	Heavy Diesel Draw-off Flow	√		
8	Stripping Steam Flow	√		
9	Heavy Naphtha 95% Distillation		√	
10	Kerosene Flash Point		√	
11	Kerosene 95% Distillation		√	
12	Light Diesel 95% Distillation		√	
13	Heavy Diesel 95% Distillation		√	
14	Kerosene Stripper Level Control Valve Opening		√	
15	Light Diesel Stripper Level Control Valve Opening		√	
16	Heavy Diesel Stripper Level Control Valve Opening		√	
17	Atm. Column O/H Drum Level Control Valve Opening		√	
18	Atm. Column O/H Drum Pressure Control Valve Opening		√	
19	HADPA Flow			√
20	Stripping Steam Duty		√	
21	Feed to Atm. Column			√

REFERENCES

Andersen, H.W. and Kummel, M. (1992). Evaluating estimation of gain directionality – Part 2: a case study of binary distillation, *Journal of Process Control*, 2, 67-86.

Rossitier, J.A. (2003). *Model based predictive control – A practical approach*, CRC Press, London.

Qin, S.J. and Badgwell, T.A., (2003). A survey of industrial model predictive control technology, *Control Engineering Practice,* 11, 733-764.

Inferential Control of Depropanizer Column Using Wave Propagation Model

S. Gupta*, A. N. Samanta, S. Ray*****

*M. W. Kellogg Ltd, Greenford, UK UB6 0JA
UK (e-mail: Sourabh.Gupta@mwkl.co.uk)
**Indian Institute of Technology, Kharagpur, India 721 302
(Tel: 91-3222-283948 ;e-mail: amar@che.iitkgp.ernet.in)
*** Indian Institute of Technology, Kharagpur, India 721 302
(e-mail: sray@che.iitkgp.ernet.in)

Abstract: In the present work a novel inferential control strategy cascaded with a nonlinear profile position controller is employed to control the top and bottom product compositions of a simulated depropanizer column. The inferential model for estimating product compositions is developed using the wave propagation model, and the composition profile position of both rectifying and stripping section is calculated using one temperature measurement from the respective section of the column. It is found that the estimation of the end product compositions using proposed technique may lead to an offset. The accuracy of the proposed inferential model can be further improved by providing an intermittent feedback of composition measurement in the form of an integral action. The wave propagation model of the depropanizer column is used in the Generic Model Control (GMC) architecture to design the profile position controllers.

Keywords: distillation column, nonlinear control, inferential control, profile position observer.

1. INTRODUCTION

A depropanizer column is used to separate propane from a mixture of components ranging from ethane to hexane. It is important to maintain the column product qualities on specification, to limit the negative effects of disturbances and upsets, and to reduce the switching time from one operating condition to another. An effective control strategy is therefore needed to control the depropanizer column. In this work, an inferential model based generic model control strategy is used which can handle disturbances and input uncertainties.

An inferential model is often used in process control when a measurement of the true variable being controlled is not available in real time. Reasons for the lack of real-time measurement include cost, reliability, and long analysis times or long dead times for sensors located far downstream. In these cases, an inferential model provides an estimate of the process variable, which can be used in the design of a controller to provide approximate regulation of the true variable. Tray temperatures are commonly used inferential measurements for product compositions. The temperature control is based on the assumption that the product composition can satisfy its specification when an appropriate tray temperature is kept constant at setpoint. In ideal situation, for a binary distillation column at constant pressure, the temperature at an end of the column is an indicator of the corresponding product composition. However, in case of a multi-component distillation column, tray temperatures do not uniquely determine the product composition. As a result, for these cases it is essential that an on-line analyzer or, at least, periodic laboratory analysis be used to adjust the tray temperature set point to the proper level.

In a Brosilow estimator [Weber and Browsilow (1972), Joseph and Brosilow (1978)] temperatures and flow rates were used for estimating unmeasured disturbances and then the derived disturbance values were used to estimate the product compositions. This estimator is based on a linearized process model. Mejdell and Skogestad (1991a, b) found that the steady state Brosilow estimator was very sensitive to modeling error for the ill-conditioned plant. In the last few decades, the development of composition estimators using partial least squares (PLS) regression have been proposed [Kresta et al. (1994)]. Furthermore, Mejdell and Skogestad dealt mainly with binary distillation columns. For a multicomponent column, tray temperatures do not correspond exactly to the product compositions. Mejdell and Skogestad have shown that the performance of the steady state PLS model for a multicomponent column is worse than that for a binary column. From the results, Mejdell and Skogestad seem to indicate the necessity of a dynamic regression estimator, which was implemented by Kano et al. (2000) in the form of dynamic partial least squares regression.

Gilles et al. (1980) reported the presence of a temperature front within a small area of the column in their extensive experimental study and showed that the locus of the temperature front is related to the product compositions. Gilles and Retzbach (1983) and Marquardt (1988, 1989) characterized the nonlinear behavior of distillation columns by the propagation of concentration profile (C-profile) and temperature profile (T-profile) in the column sections. Lang and Gilles (1990) presented an estimation technique that can be applied to complex processes in chemical industries. Adapting well advanced theories of fixed bed adsorption,

Hwang (1991) proposed a nonlinear wave theory for distillation columns which views the movement of composition and temperature profiles as nonlinear waves. They reported that these waves tend to sharpen for most situations and become constant pattern waves. Han and Park (1993) proposed a model based composition controller design incorporating Hwang's nonlinear wave model into the generic model control (GMC) framework of Lee and Sullivan (1988). To overcome the difficulty of composition measurements, Shin et al. (2000) proposed a C-profile position observer based on the temperature measurements. However, their proposed profile position estimation algorithm is applicable only for a binary system. Recently Gupta et al. (2009) has extended the application of profile position control to a debutanizer column.

In this work distillate and bottom propane compositions of a simulated depropanizer column are controlled by using composition to C-profile position cascaded controllers (Figure 1). The compositions are inferred from the C-profile position observer using one temperature measurement from each section (rectifying/ stripping). The objective of this article is to present a new approach to infer and control product compositions using the C-profile position of a depropanizer column, which uses one temperature measurement from the respective (rectifying/stripping) section.

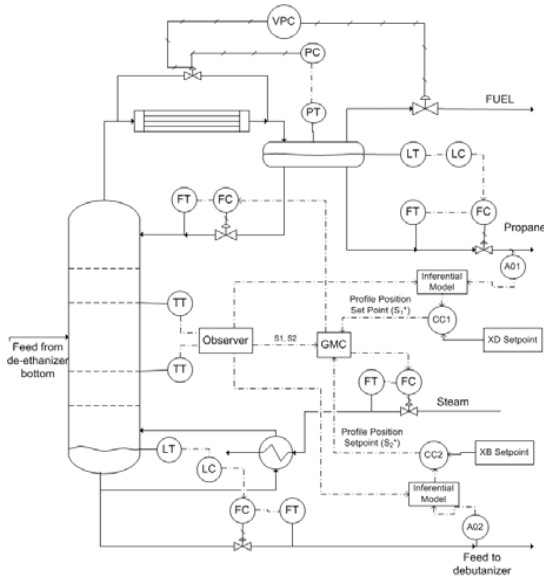


Figure 1: Depropanizer column with control strategy

2. PROCESS DESCRIPTION AND CONTROL STRATEGY

The depropanizer column of a gas recovery unit is simulated in this work [Huang and Riggs (2002)]. The depropanizer column consists of 40 trays, and feed, a mixture of C_2-C_6 components, is fed to the column at 22nd tray (counted from the bottom). The column has a partial condenser and the pressure is controlled via a hot vapor bypass around the overhead condenser. The distillate accumulator level is controlled by adjusting propane product flow rate. In a depropanizer column, the control objective should be to remove impurities (C_4+ components) in the distillate and maintain minimum possible propane loss in the bottom product to maximize the yield of propane in the distillate. This is a separate optimal control problem and is not in the scope of this work. Here the control is achieved by controlling the propane compositions in the distillate and bottom product to their already known optimum targets. The control scheme is shown in Figure 1 and the nominal values required for the distillation column simulation is presented in Table 1. The composition controllers (CC1 & CC2) are PI controllers which generate the profile position setpoints by using the inferred values of propane composition from the inferential model.

reflux rate (k mol/sec)	0.30		
reboiler duty (k joule/sec)	5248.8		
condenser duty (k joule/sec)	-4881.7		
Stream Details			
	feed	*distillate*	*bottoms*
flowrate (k mol/sec)	0.21	0.06	0.15
Temperature (°C)	86	43	112
pressure (k pascal)	3052	1515	1612
Composition (mol %)			
C_2	0.6	2	-
C_3	30	95.1	1.2
C_4	54.2	2.9	76.8
C_5	8.1	-	11.7
C_6	7.1	-	10.3

Table 1: Operating variables for depropanizer column

3. NONLINEAR WAVE MODEL

The dynamic behavior of distillation columns is characterized by the propagation of concentration or temperature profile in the column sections. Numerical simulation results of this typical dynamic behavior for the depropanizer column presented in figure 2 for 10% heavier and 10% lighter feed (Table 2). Propane composition and temperature profile moves up or down to the column ends as a result of increase or decrease of heavier components in the feed. It is also evident from the figure that both the waves ultimately tend to become steep and constant pattern as they move up or down the column.

The travel of such a constant-pattern self-sharpening wave can be characterized by the 'shock wave' velocity [Hwang (1991)] tracking the propagation of specific value of concentration. This wave velocity is derived from the material balance across the wave:

$$u_{\Delta} \equiv \left(\frac{\partial \sigma}{\partial \tau}\right)_{\Delta} = \frac{V}{F} \cdot \frac{\Delta y / \Delta x - L/V}{1 + r\left(\Delta y / \Delta x\right)} \qquad (1)$$

where r is vapor to liquid holdup ratio, τ is normalized time ($\tau = tF / NM$), and σ is normalized distance from bottom of the column ($\sigma = k / N$). Assuming the liquid flow is so slow that local equilibrium is attained, y in equation (1) can be

substituted with the vapor liquid equilibrium relation. The concentration and temperature waves will travel to either one of the column ends unless the balance of convective transports is carefully maintained to have a zero shock wave velocity with the compositions and flow rates of all streams entering the column sections including feed, reflux, and reboiler vapor flow. Therefore, the behavior of the column is severely nonlinear and sensitive since even a small upset of the balanced condition will lead to a large shift of the composition/temperature profile, giving dramatic changes in the product purity. By analyzing the profile positions for each section and the compositions at the column ends (distillate/bottoms) a model equation can be obtained to correlate the profile position with the compositions from the steady state data for the profile position and the top/bottom compositions collected from the steady state plant model simulation.

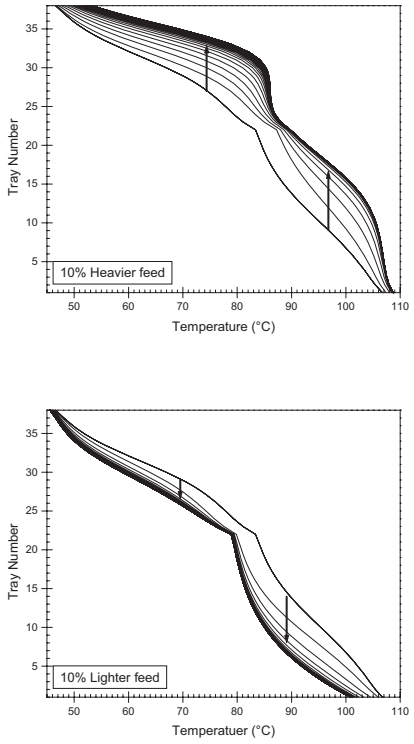


Figure 2: Dynamic profiles of the depropanizer column to a step disturbance (10% heavier feed and 10% lighter feed) of feed composition in open loop (each curve is separated by 5 minutes).

Feed composition

Composition (mol %)	normal	10% lighter	10% heavier	20% heavier	30% heavier
C2	0.6	0.64	0.56	0.52	0.48
C3	30	32.12	27.96	25.99	24.10
C4	54.2	52.51	55.83	57.39	58.90
C5	8.1	7.85	8.34	8.58	8.80
C6	7.1	6.88	7.31	7.52	7.72

Table 2: Feed composition in different scenarios

4. DEPROPANIZER CONTROLLER DESIGN

4.1. Profile position controller

The profile position controller is a nonlinear model-based controller, and is designed by embedding a nonlinear wave model directly into the generic model control (GMC) control framework.

The GMC equation can be written as the following in the case that state vector is a composition profile position S.

$$\frac{dS}{dt} = K_1\left(S^* - S\right) + K_2 \int_0^t \left(S^* - S\right) dt' \tag{2}$$

where, S and S^* are the profile position and its setpoint respectively; dS/dt is the propagation rate of profile. S is expressed in terms of the normalized distance from the bottom of the column ($S = 0$ at the bottom; $S = 1$ at the top). The propagation rate can be expressed from the nonlinear wave model as follows:

$$\frac{dS}{dt} = u = \frac{V}{F}\frac{\Delta y/\Delta x - L/V}{1 + r\left(\Delta y/\Delta x\right)} \tag{3}$$

Distillation columns, in general, have two sections: one is the rectifying section and other is the stripping section. Combining equations (2) and (3) gives one equation for each section as follows:

$$\frac{V}{F}\frac{\Delta y/\Delta x - L/V}{1 + r\left(\Delta y/\Delta x\right)} - K_{11}\left(S_1^* - S_1\right) - K_{12}\int_0^t \left(S_1^* - S_1\right) dt' = 0 \tag{4}$$

$$\frac{\bar{V}}{F}\frac{\Delta \bar{y}/\Delta \bar{x} - \bar{L}/\bar{V}}{1 + r\left(\Delta \bar{y}/\Delta \bar{x}\right)} - K_{21}\left(S_2^* - S_2\right) - K_{22}\int_0^t \left(S_2^* - S_2\right) dt' = 0 \tag{5}$$

where, subscripts 1 and 2 represent rectifying section and stripping section respectively, and L and V are the liquid and vapor flow rates respectively in the rectifying section and $\bar{L}$ and $\bar{V}$ in the stripping section. The profile position and the slope of the equilibrium curve at the representative concentration can be estimated by the profile position observer. Mass balance around the feed tray gives

$$\bar{L} = L + qF \tag{6}$$

$$V = \bar{V} + \left(1 - q\right)F \tag{7}$$

where, q is the liquid mole fraction of the feed. Knowing the feed conditions L, V, $\bar{L}$ and $\bar{V}$ is calculated from equations (4)-(7).

4.2. Online estimation of the profile position

The success of the inferential controller is mainly dependent on the ability to estimate the profile positions for both rectifying and stripping sections. The profile position in each section can be regarded as the location of the constant pattern wave representing a single point corresponding to a representative temperature. The profile position of the constant pattern wave can be determined by tracking the representative temperature instead of the entire wave.

In this case, the profile position observer is designed using the nonlinear wave model for the depropanaizer with an

additional feedback of weighted output error of temperature(s) feedback.

$$\dot{S} = \frac{dS}{dt} = \frac{V}{F}\frac{\Delta y/\Delta x - L/V}{1 + r\left(\Delta y/\Delta x\right)} + \sum_{i=l}^{m} k_1\left(T_i\right)\left(T_i - \hat{T}_i\right) \quad (8)$$

$$\frac{\Delta y}{\Delta x} = \frac{\dot{S} + L/F}{V/F - r\dot{S}} \quad (9)$$

$$\hat{T}_i = k_2\left(S_i - S\right) + T_s \quad (10)$$

$$k_1\left(T_i\right) = k_0 \exp\left[-b\left(T_i - T_s\right)^2\right] \quad (11)$$

where, i is the measurement tray number and l and m are the number of first measurement tray and the number of last measurement tray in a column section respectively. For $\Delta y/\Delta x$ and T relationship the steady state plant data can be used which will be associated with the tray efficiencies also. The tray temperature measurement location (26th and 18th tray) in each section is selected based on the inflection points in the temperature waves. The above equations are solved with initial estimates of the profile position S and representative slope $\Delta y/\Delta x$ to obtain the profile position. The sample time for the composition controller has been taken as 2 seconds while that of the profile position controller was 0.2 seconds.

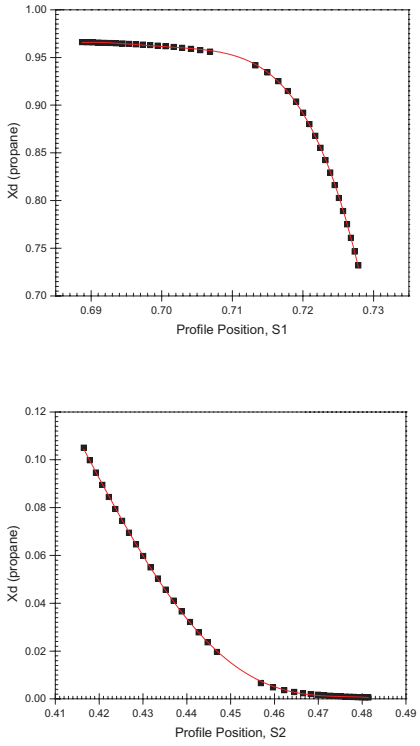


Figure 3: Graphical representation of steady state data set, of rectifying section profile position vs X_D (propane), and stripping section profile position vs X_B (propane).

5. INFERENTIAL CONTROL

In earlier work, the column end compositions were estimated using more than one measurement in the form of temperature, flow rate, and heat duty etc. In the proposed estimator one temperature measurement from each section (rectifying/ stripping) is used to find out the distillate and bottom composition of propane in the depropanizer.

5.1. Model identification using profile position

In this section we are proposing a model based inferential control using the profile position estimation. By analyzing the profile positions for each section and the compositions at the column ends (distillate/bottoms) a model equation can be obtained to correlate the profile position with the compositions. The steady state data for the profile position and the top/bottom compositions collected from the steady state plant model. Figure 3 shows that the data obtained from the plant model can be expressed by a 3rd order polynomial.

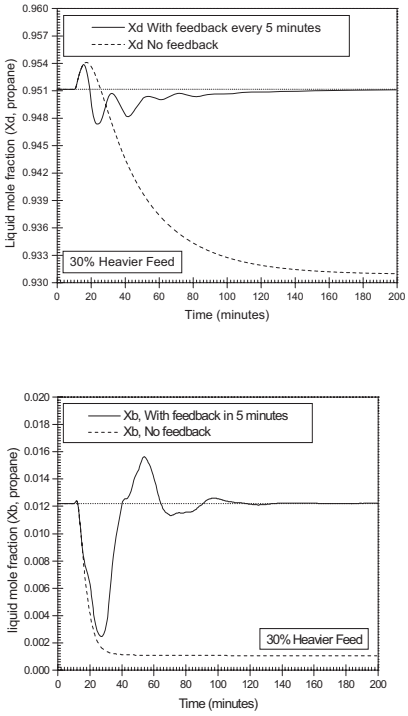


Figure 4: Closed loop transient response of the distillate composition (propane) and bottoms composition (propane) for 30% heavier feed using inferential control, with feedback /no feedback to the inferential composition estimator for the depropanizer column

Figure 4 shows that the proposed model is fair enough to control the end compositions of the depropanizer column. However, the current structure of the model leads to an offset with the final setpoint compositions.

5.2. Model identification using profile position with feedback

Our study leads that the composition measurements are required for the tight control of the end compositions of the depropanizer column. To remove the offset, an integral action

as a feedback to the estimator is proposed. The final form of the composition estimator can be expressed as follows:

$$x^{est.} = x^{\text{model}} + k' \int \left(x^{\text{last measurement}} - x^{\text{model}} \right) dt \quad (8)$$

where, k' is a tuning parameter. Figure 4 shows that the proposed model with feedback is able to control the end compositions of the column without any offset. For the depropanizer column value of tuning parameter k' is tuned at 1 and 0.1 for rectifying section and stripping section respectively, and a lag of 5 minutes allowed for the composition measurement.

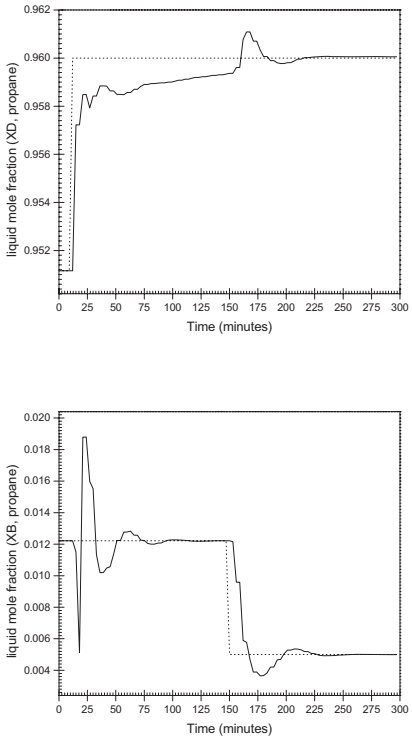


Figure 5: Closed loop transient response for the distillate and bottoms composition (propane) for setpoint change in X_D (propane) of 0.009 (0.951 to 0.96) at 10 minutes followed by a setpoint change in X_B (propane) of 0.007 (0.012 to 0.005) at 150 minutes, with feedback to the inferential composition estimator

6. RESULTS AND DISCUSSION

The depropanizer column with all control loops is simulated to verify the proposed control strategy.

6.1. Effect of composition setpoints change

Two concurrent setpoint changes are implemented to the top and bottom composition controllers. Closed loop transient response for the distillate and bottoms composition (propane) for a setpoint change in X_D (propane) of 0.009 mole fraction (0.95 to 0.96) at 10 minutes followed by a setpoint change in X_B (propane) of 0.007 mole fraction (0.012 to 0.005) at 150 minutes is shown in Figure 5. Increasing the propane purity in the distillate caused an immediate loss in propane content in the bottom causing a shift in the profile position. To maintain the propane composition in the bottom product the GMC controller has to put back the profile position in place which causes a sluggish response in the top composition response after the initial jump. When the bottom composition setpoint is also changed, both the controllers acted speedily because the changes are in favored direction from the viewpoint of the process dynamics.

6.2. Effect of noise and input uncertainty

To examine the robustness aspects of the controller about input uncertainty and temperature measurement noise, a simulation experiment is conducted on the depropanizer controlled by the proposed control strategy. During the simulation experiment following setpoint changes and disturbances are given to the system:

Setpoint change in X_D (propane) at 10 minutes (from 0.95 to 0.96).

Setpoint change in X_B (propane) at 150 minutes (from 0.012 to 0.01).

Step disturbance in the feed composition at 300 minutes (20% heavier feed, Table 2).

The following uncertainties have been taken into account during the simulation experiment:

Random disturbance in the feed flowrate (±5%)

Random disturbance in the reboiler heat duty (±10%) and reflux rate (±10%)

Temperature measurement noise (±0.5 °C)

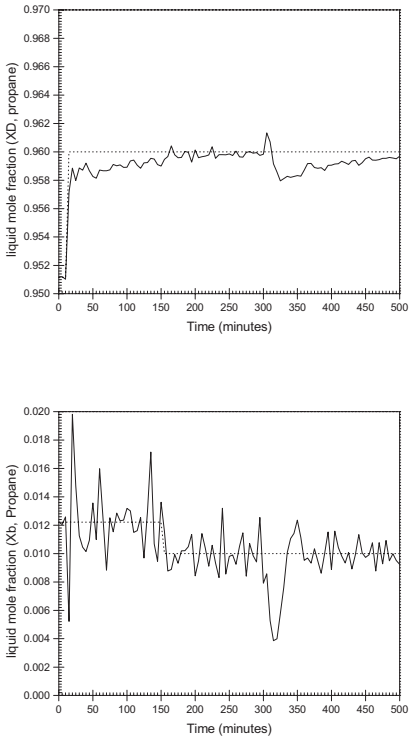


Figure 6: Effect of noise and input uncertainty: closed loop transient response in X_D (propane) and X_B (propane)

The distillate and bottom propane composition responses are shown in Figure 6. In spite of the severe input disturbances and measurement noise, the proposed controller is able to control the product propane compositions with reasonable speed of response and accuracy.

7. CONCLUSIONS

This study has shown a method of developing an inferential model for process control of depropanizer column using the observed profile position. A nonlinear profile position observer has also been developed to estimate the profile position of the column section with sufficient accuracy using temperature measurement. The profile position has been shown to be a powerful approach to building such models and uses the existing available measurements in the depropanizer. However, steady state plant data is needed for design of such models which may be collected while the process is operating under a feedback structure. Under a process/model mismatch, cascaded inferred composition to nonlinear profile position controller performed adequately well in controlling the depropanizer column. It will be interesting to compare the performance of the proposed controller with other nonlinear controller using input-output linearization controller or nonlinear model predictive controller which are much more computationally intensive than the proposed controller.

8. REFERENCES

Weber, R. and Brosilow, C. B. (1972), The use of secondary measurements to improve control. AIChE Journal, 18(3), 614-623.

Joseph, B. and Brosilow, C. B. (1978), Inferential control of processes. AIChE Journal, 24 (3), 485-509.

Mejdell, T. and Skogestad, S. (1991a), Estimation of distillation compositions from multiple temperature measurements using partial least squares regression. Industrial & Engineering Chemistry Research, 30, 2543-2555.

Mejdell, T. and Skogestad, S. (1991b), Composition estimator in a pilot plant distillation column using multiple temperatures. Industrial & Engineering Chemistry Research, 30, 2555-2564.

Kresta, J. V., Marlin, T. E. and MacGregor (1994), J. F., Development of inferential process models using PLS. Computers & Chemical Engineering, 18(7), 597-611.

Kano, M., Miyazaki, K., Hasebe, S. and Hashimoto, I. (2000), Inferential control system of distillation compositions using dynamic partial least squares regression. Journal of Process Control, 10, 157-166.

Gilles, E. D., Retzbach, E. and Silberberger, F. (1980), Modeling, simulation, and control of an extractive distillation column. Computer Applications to Chemical Engineering, 481-491.

Gilles, E. D., and Retzbach, B. (1983), Reduced models and control of distillation columns with sharp temperature profiles. IEEE Transactions on Automatic Control, 28 (5), 628-630.

Marquardt, W. (1988), Nonlinear model reduction for binary distillation. IFAC Symposium DYCORD ’86-Dynamics and Control of Chemical Reactors and Distillation Columns, Bournemouth, UK. Pergamon Press, Oxford, 123-128.

Marquardt, W. (1989), Concentration profile estimation and control in binary distillation. IFAC Workshop Model-Based Process Control, Atlanta, USA. Pergamon Press, Oxford, 137-144.

Lang, L. and Gilles, E. D. (1990), Nonlinear observers for distillation columns. Computers & Chemical Engineering, 14(11), 1297-1301.

Hwang, Y. L. (1991), Nonlinear wave theory for dynamics of binary distillation columns. AIChE Journal, 37 (5), 705-723.

Han, M. and Park, S. (1993), Control of high-purity distillation column using a nonlinear wave theory. AIChE Journal, 39(5), 787-796.

Lee, P. L. and Sullivan, G. R. (1988), Generic Model Control (GMC). Computers & Chemical Engineering, 12(6), 573-580.

Shin, J., Seo, H., Han, M. and Park, S. (2000), A nonlinear profile observer using tray temperatures for high-purity binary distillation column control. Chemical Engineering Science, 55, 807-816.

Gupta, S., Ray, S. and Samanta, A. N. (2009), Nonlinear Control of debutanizer column using profile position observer. Computers and Chemical Engineering.

Huang, H. and Riggs, J. B. (2002), Comparison of PI and MPC for control of a gas recovery unit. Journal of Process Control, 12, 163-173.

Advanced Process Control Wide Implementation in Alunorte Digestion Unit

Rafael Lopes*, Leonardo Vieira*, Ayana Oliveira, Jedson Santos**, Márcia Ribero**, Jorge Aldi**, Jorge Charr***,**

** Honeywell do Brasil. Av. Tamboré, 576 - Barueri – São Paulo – Brazil (e-mail: rafael.lopes@honeywell.com)*
*** Alunorte - Alumina do Norte do Brasil. Rodovia PA 481 km 12 – Distrito do Murucupi Barcarena – Pará – Brazil (e-mail: jorge.aldi@alunorte.net)*
**** Honeywell Venezuela. Av. Ppal Los Cortijos con 4ta Transversal. Edif. Honeywell. Los Cortijos de Lourdes. Caracas – Venezuela (e-mail: jorge.charr@honeywell.com)*

Abstract: The most competitive environment generated the need of process performance optimization. Performance optimization means produce the same amount of product, more effectively and spending less money. On the alumina market, that's fundamental in this economy scenario. Robust Multivariable Predictive Control Technology becomes one of the main tools to optimize this class of plants. This paper will discuss the application and benefits of this technology to alumina digestion units, implemented in 5 interconnected digesters. This digestion interconnection is a whole digestion train, and the plant has 5 of those. The APC philosophy is based on process variability reduction, and consequently operations optimization, against plant constraints. Since alumina – caustic ratio (A/C) is the key plant variable, it has a fundamental role in this variability reduction. The main challenge in this project was to coordinate the use of 5 bauxite grinders and 2 more grinder bauxite flows to the 5 digesters. The implementation was made in 3 phases and the project length was approximately 18 months, generating more than 1.00% increase in overall production, rather than A/C variability reduction.

Keywords: Advanced Process Control, Alumina, Alunorte, Honeywell, Multivariable, RMPCT

1. INTRODUCTION

Significant economic savings can be generated to alumina plants, through the utilization of new control technologies that uses the existing infrastructure and require a reduced support team. The global market and supplier consolidation created a more competitive environment, which drives the need of production and performance optimization. The multivariable predictive control becomes one of the main tools in this scenario. This paper will discuss the application and benefits of this technology to the alumina digestion units.

The challenge to any alumina refinery is to minimize the cost of production per tonne of alumina, consistently with safety and environmental considerations. It is translated into alumina production maximization (plant flow and yield) and energy costs per tonne of alumina minimization.

In this scenario, the digestion process is the one that has the biggest potential to the robust multivariable predictive control technology (RMPCT) implementation. Rather than this, the digestion is considered by most of refineries as a key-unit to the production and also is the one that offers the best data for an APC modelling.

1.1 Process Description

The process for obtaining alumina from bauxite ore was developed and patented by Karl Josef Bayer in 1888. Typically, depending on the quality of the ore, between 1.9 and 3.6 tonnes of bauxite are required to produce 1 tonne of alumina.. Bayer process is cyclical and involves many unit operations, like digestion, solid – liquid separation and crystallization.

Overall, bauxite ore is digested in caustic solution concentrate in temperatures ranging from 145 to 270°C, depending on the nature of the ore. Under these conditions, most mineral species that contains aluminum is dissolved, forming sodium aluminate, soluble, as shown in equations (1) and (2).

$$Al(OH)_3 + NaOH \rightarrow NaAlO_2 + 2H_2O \quad (1)$$

$$AlO(OH) + NaOH \rightarrow NaAlO_2 + H_2O \quad (2)$$

The portion of the ore that is insoluble in caustic solution after digestion (red mud) is removed by sedimentation and filtration process. The pregnant liquor in alumina is send to the precipitation, which is almost pure crystals of $Al(OH)_3$. The hydrate precipitate is removed, washed and sorted. Alumina is then obtained by their calcination.

1.2 RMPCT (Robust Multivariable Predictive Control Technology)

RMPCT technology represents an advance of the traditional MPC technologies. Like the others, this technology models the process, calculate the necessary predictions and use multivariable control movements in order to: optimize the process, maintain the variables inside operational limits and respect the process and plant constraints. The performance gain and robustness is due to a feature called "range control algorithm" (RCA), which makes that the disturbances and prediction errors inherent to the process are considered in the future movement plan. Figure 1 sketches how the RCA technology works.

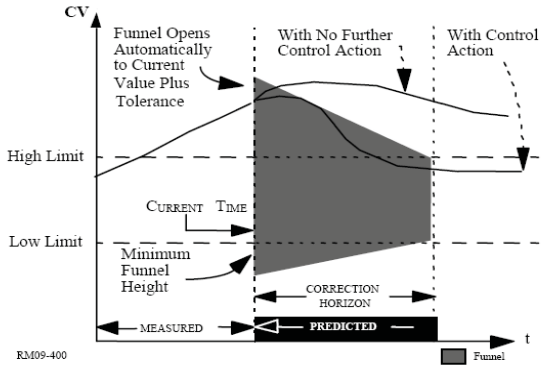


Figure 1 - RCA Technique Controlling a CV Inside Limits

The correction horizon concept is that CV errors are reduced to zero at the correction horizon in the future. Prior to the correction horizon, the controller is free to determine any trajectory for the CV as long as the CV is brought within limits or to setpoint at the correction horizon. Because no trajectory is imposed on the controller, the controller has the freedom to determine a trajectory that requires minimum MV movement and is least sensitive to model error.

However, the correction horizon by itself does not say anything about what happens to the CV prior to the horizon. It is important that the controller does not transiently move a CV farther outside a limit while correcting other CV errors, even though all CVs are brought to zero error by their correction horizons. Limit funnels are used to prevent the controller from introducing transient errors prior to the correction horizons, by defining constraints on the CVs that are imposed at intervals from the current interval out to the horizon.

These features drives the application to deal smoother and more efficiently with model mismatches (gain inversion, colinearities, bigger or smaller gains than the real, dynamic errors). Rather than this, the tuning in this technology is based on the controlled variables and not in the manipulated variables.

2. APPLICATION OF RMPCT IN ALUNORTE DIGESTION UNIT

2.1 Digestion Process Description

The digestion unit is designed to extraction alumina from bauxite using caustic solution in high temperature and remove dissolved silica from the liquor leaving the digesters to ensure product hydrate of the desired quality.

The alumina extraction is carried out in a train consisting of five vertical digesters arranged in series. The first step is to dissolve a most part of alumina in ore mixing slurry bauxite and heater spent liquor, in small digesters, equipped with agitators. The large digesters, without agitators, in series are to keep the residence time to reduce the silica dissolved by desilication reaction to a tolerable level.

The five vertical digesters are sized to provide a total of 60 minutes nominal retention time. Varying the liquor outlet temperature from the second live steam heater controls the digestion temperature. In occasion when one digester is taken out of operation the temperature is increased approx 1°C to compensate.

2.2 General Control Strategies

The Advanced control strategies of the Bayer plant are used to control blow off ratio, caustic concentration and to keep productivity and quality.

Alumina refineries generally operates with advanced A/C ratio control systems, involving feed forward with feedback trim and utilizing on-line measurement of liquor properties, such as electrical conductivity and density.

In 2006, ALUNORTE concluded the project of expansion 2 with five lines, in operate, with total liquor flow of 5610 m3/hr and installed capacity of 4.3 Mt/year.
The project to implement RMPCT control is divided in three phases:

- Phase I : Implementation of control on digestion 3.
- Phase II : Implementation of control on digestions 1 and 2.
- Phase III: Implementation of control on digestions 4 and 5.

2.3 Controller Objectives

The advanced control objectives for the digestion section are described below:

- Control A/C ratio to operator specified target
- Maximize productivity (bauxite and liquor flows), subject to process constraints
- Provide safe and stable operation
- Protect the unit when possible from defined, measurable constraints such as hydraulic, mechanical and environmental constraints.

2.4 Application Methodology

The RMPCT implementation consisted on the following steps:
Data and information gathering → Pre-Step Test → Step Test → Mathematical Modeling → Installation and sustaining

The implementation methodology is detailed bellow:

- Collection of: historical data, operation screens, process flow diagrams, engineers and operators information by interviews;
- Instrumentation review, control strategy setup and related loops tuning;
- After the analysis of all data, a preliminary controller design matrix is defined and discussed. This matrix will drive the initial plant tests (Pre-Step test);
- Prior to starting a test, the process and control system must be brought to a suitable starting condition, and allowed to settle if any changes were made. This will involve ensuring that the process is away from limits or "wind-up" conditions, and making sure that all control loops are in the correct modes.
- Pre-Step Testing is necessary to determine the steady state gain and settling times to be able to conduct precise Step Testing. After analyze of the collected data, final decisions of controller structure and step size will be issued in a report that acts as the basis for the formal step testing
- After the Pre-Step test, the Step Test is performed, applying steps to the considered manipulated variables. The steps are applied with variable time and amount, in order to indentify the actual interactions that will build the definitive multivariable control matrix.
- Using the data gathered on the Step Test, the models are constructed and the RMPCT is built. The matrix is validated, analyzing the predictions and controller offline simulation.
- After the matrix and control construction, the software connections with DCS are configured and an initial software tuning is performed.

2.5 Basic Controller Structure

The main manipulated variables are:
- Bauxite slurry flow
- Liquor flow
- Steam flows of relevant plant heat exchangers

The main controlled variables are:
- Alumina/Caustic Amount Ratio
- Digestion Conditions (temperatures, pressures and volume controls)
- Feed to digestion conditions.

The following table represents the controller gain matrix. MVs 1 to 4 refer to the unit mass balance variables. MVs from 5 to 7 refer to the unit energy balance variables. CVs from 3 to 6 refer to the unit energy balance variables. Other CVs are related to the unit mass balance parameters.

Table 1 – RMPCT Gain Matrix to the Digestion

	MV1	MV2	MV3	MV4	MV5	MV6	MV7
CV1	+	+	-	-			
CV2	-	-	-	-			
CV3	-	-	-	-	+	+	
CV4	-	-	-	-	+	+	
CV5	-	-	-	-	+	+	
CV6	+						
CV7		+					
CV8			+				
CV9				+			
CV10					+		
CV11							
CV12							
CV13						+	
CV14							-
CV15	+	+					

For example, MV1 is the bauxite flow and CV1 is the Alumina – Total Caustic ratio. If the bauxite flow is increased, the A/TC ratio is increased, after a dead time. CV2 is the residence time and the CV's 3, 4 and 5 are digestion temperatures. It can be noticed that, if the bauxite flow is increased, the residence time on the digestion system decreases and the digestion temperature decreases, too. In this controller, the A/TC ratio is controlled on a target, not inside operational limits.

MV3 is the liquor flow to the digestion system. This variable has a significant influence on the A/TC ratio as is one of the main handles for it. For the digestion temperature and residence time, it can be expected the same behaviour as on the interaction between the bauxite MV and the same controlled variables.

CV's from 6 to 13 are valves and they are controlled as constraints on the controller.

3. MODELLING RESULTS

3.1 Modelling Achievement

The historical data gathered was enough to get good models to build the control matrix. To the mass balance variables, around 10 steps were used and to the energy balance variables, around 6 steps were used. This difference is due to the bigger relevancy of the mass balance, since the main variable (A/C) is influenced by this group of variables.

The following figure represents one of the models between the manipulated and controlled variables. In this case, the model represents the behaviour of the Alumina – Caustic Concentration ratio, against on of the mass balance manipulated variables.

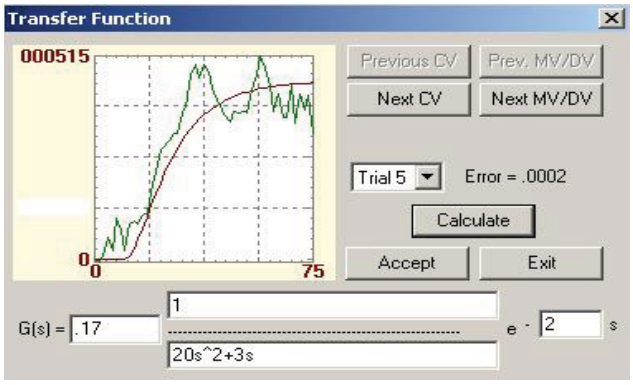


Figure 2 – Model Between Bauxite and the Alumina-Caustic Concentration Ratio (CV)

The unit studied has some valve opening problems on the liquor and pulp heating section. These problems are due plugging, caused by the material that goes inside the heat exchanger tubes. In order to minimize this problem, the valves were modelled, against their steam flows. A model example is showed on the Figure 3.

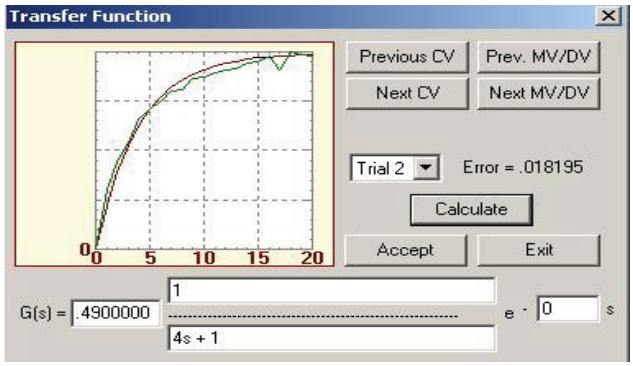


Figure 3 – Valve modelling against the steam flow

3.2 Model Validation

After the modelling, the predictions were analyzed, in order to check if the model is coherent with the real process data, found on the plant test. This validation is one of the last steps, before the controller implementation. The following pictures show the prediction results.

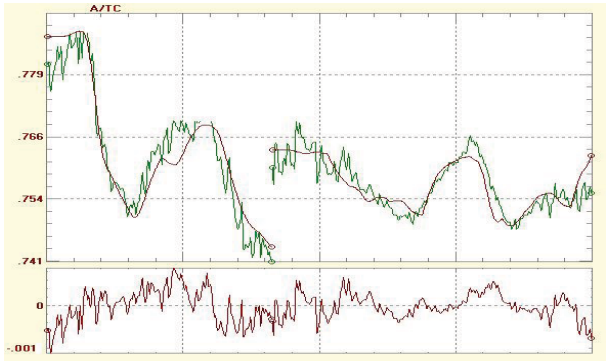


Figure 4 –A/TC Prediction

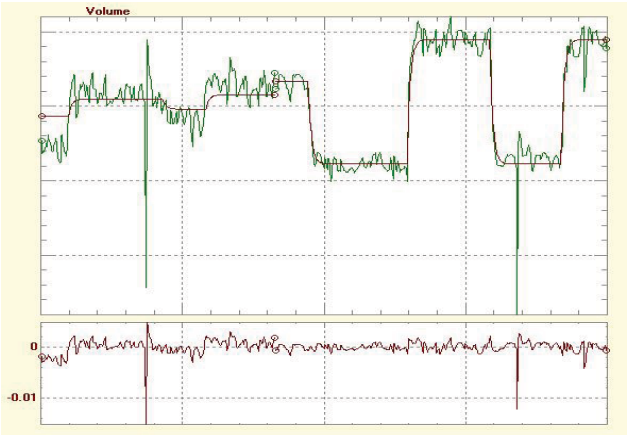


Figure 5 – Digestion Volume Prediction

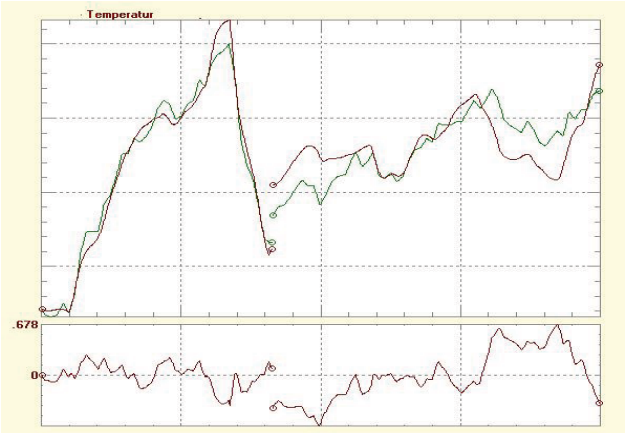


Figure 6 – Temperature Prediction

The Figures show good prediction results. Thus, the proposed and modelled matrix could be tested on the offline controller simulations. In the simulation mode, the control strategies and controller tuning are tested. Rather than this, the controller behaviour against critical situations can be validated. After this last validation, the controller was ready to be implemented on this alumina digestion unit

4. IMPLEMENTATION RESULTS

4.1 Overview

The main reason to implement RMPCT in digestion unit is to keep A/C control at the set point, decrease variability of the system, to increase digestion yield and to keep safety and stable operational conditions.

The main controllers are: DG4B_CLT (Digestion 3 controller). The others controllers are called: DG4A1_CLT, for digestion 1 and DG4A2_CLT, for digestion 2, DG4C1_CTL, for digestion 4 and DG4C2_CTL for digestion 5.

In order to evaluate digestion operation results with RMPCT, it's necessary to consider two parameters:

- Digestion Blow Off (DBO) ratio;
- Digestion Yield

4.2 Digestion Blow off (DBO) ratio

DBO ratio is the main parameter to determinate digestion yield. A good control of this parameter means smaller variability, which allows a higher yield at the digestion outlet. For digestion lines 1,2 and 3, the set point DBO ratio is 0,750. For the digestion lines 4 and 5, the set point DBO ratio is 0,759.

4.3 Digestion Yield

To calculate the digestion yield, the equation 3 is used:

$Y = (((C_{SL}\text{-}S_l)\ .\ A/C_{DBO})\text{-}(C_{SL}.A/C_{SL}))\text{-}C_{STT}$ (3), where:

CSL = Spent liquor caustic concentration (g/l)
SL = Silica lost
A/CDBO = Digestion blow off ratio
A/CSL = Spent liquor ratio
CSTT = Spent liquor solids concentration

5. RMPCT PERFORMANCE

5.1 DG4B_CTL

Controller performance was evaluated through the comparison between two different periods of digestion 3 operation. Those periods represent the time when RMPCT was turned on and off.

In the digestion operation without RMPC, the average DBO standard deviation was 0,005 higher, when compared with the digester operation with RMPC (0,002). It represents that the controller performed satisfactorily. Table 2 shows draft of DBO ratio performance with and without RMPCT operation.

Table 2- Draft DBO ratio performance

RMPCT Operation	Average DBO Ratio	Average δDBO
Controller off	0,748	0,005
Controller on	0,751	0,002

This improvement on the DBO represented a gain of 1,02% on the digestion yield. The total time spent for this project phase was 8 months.

Figure 9 and 10 show DG04B_CLT performance when RMPCT was on and off. When RMPCT was on, A/C values were more stable than when the controller was off. A DBO standard deviation of 0,003 was achieved when the Profit Controller was turned on, instead of 0,006, when controller was off.

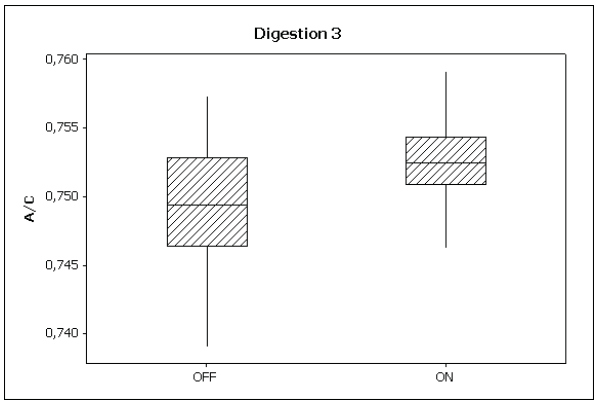


Figure 7 – Digestion 3 A/C when RMPCT on and off

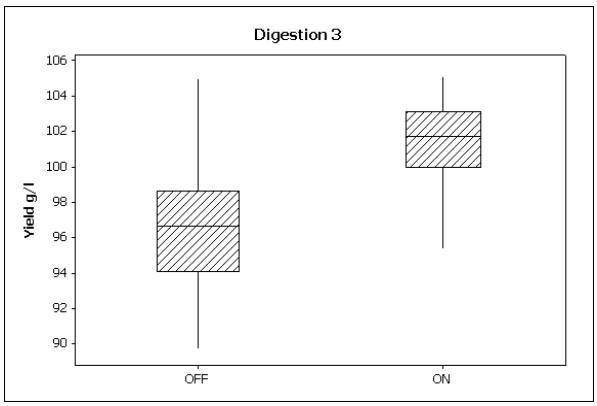


Figure 8 – Yield Digestion 3 when RMPCT on and off

5.2 DG4A1_CTL and DG4A2_CTL

RMPCT for digestion 1 and 2 was evaluated, in order to define a gain with controller in these units. The evaluation followed the same methodology as in the digestion 3.
Figure 11 shows behaviour of A/C in periods when RMPCT is off and on, with average of 0.749 and 0.753, respectively. Standard deviation of the DBO in periods where RMPCT was turned on was better than when the controller was turned off, with averages of 0,02 and 0,01, respectively.

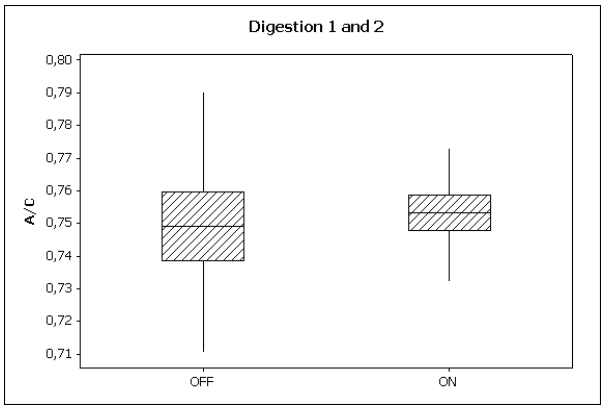


Figure 9 – Digestion 1 and 2 A/C when RMPCT on and off

Figure 12 shows an yield digestion gain of 1,85%. The yield when the controller was turned off was 98,93 g/l and when the controller was turned on was (100,77 g/l).

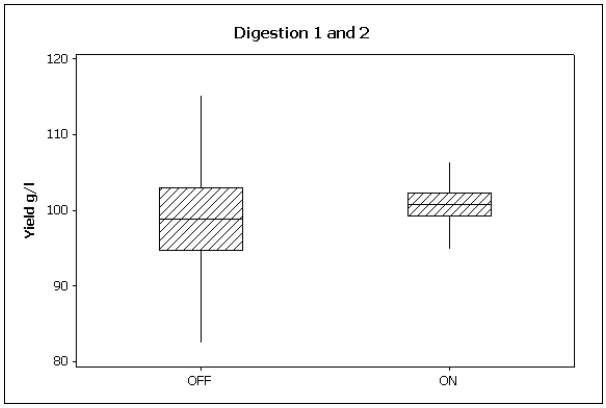


Figure 10 – Yield Digestion 1 and 2 when RMPCT on and off

5.3 DG4C1_CTL and DG4C2_CTL

RMPCT for digestion 4 and 5 was evaluated, in order to measure the gain obtained with its implementation. The evaluation follows the same methodology as in the other digestion trains.

Figure 13 shows behaviour of A/C in periods when RMPCT is off and on, with average of 0.756 and 0.759, respectively. Standard deviation δDBO in periods where RMPCT is on is better than it's off, with average of 0,02 and 0,01.

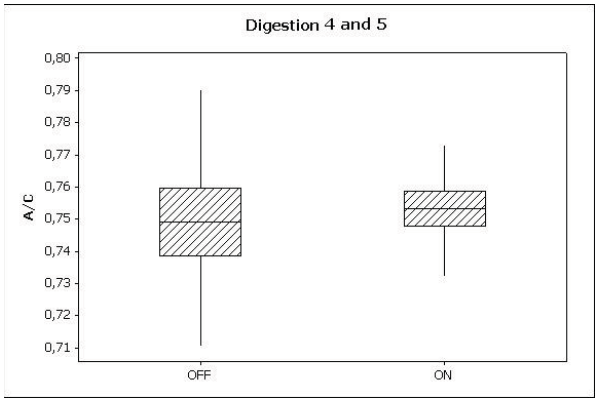


Figure 11 – Digestions 4 and 5 A/C when RMPCT on and off

Figure 14 shows yield digestion gain of 1,69%. The yield when the controller was turned off was 99,72 g/l and when the controller was turned on was 101,44 g/l.

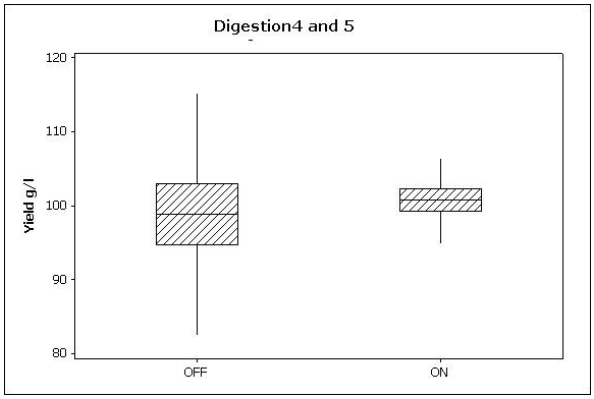


Figure 12 – Yield Digestion 4 and 5 when RMPCT on and off

6. CONCLUSION

A good RMPCT implementation on Alumina Digestion was described. The A/TC variability reduction and a bigger operation stability were proven. Also, the opportunity to operate the plant close to the operational constraints represents a productivity increase and a plant debottlenecking. The steam and liquor consumption didn't change significantly, since the objective was to use the debottlenecking to increase alumina production.

Nevertheless, in this application the liquor flow was maintained constant, due to operational restrictions. If the liquor could move, probably the results would be better than the achieved. Other source of improvement (which wasn't explored in this work) is the increase of operator training on this tool. Since the operators are the heaviest users of the system, training them to help on the optimization, pushing constraints and widening the operation limits can generate a bigger production improvement, than it was achieved.

REFERENCES

Garcia, C.E., D.M. Prett and M. Morari (1989). "Model predictive control: Theory and practice - a survey". Automatica 25(3), 335–348.

Qin, J., Badgwell, T., "An Overview of Industrial Model Predictive Control Technology". Department of Chemical Engineering, Rice University, USA (1997)

Lopes, R., Charr, J. ,"Alunorte Digestion Phase I Detail Design V2.1", Belém, Brasil (2007).

Ribeiro, M., "Benefícios Encontrados com Implementação RMPCT Honeywell", Belém, Brasil (2007).

Den Hond R., Hiralal I., Rijkeboer A., "Alumina Yield in the Bayer Process; Past, Present and Prospects". TMS (The Minerals, Metals & Materials Society), 2007.

www.solucoesavancadas.com.br – More advanced solutions business cases

Dynamic Models and Open-Loop Control of Blood-Glucose for Type 1 Diabetes Mellitus

Hsiao-Ping Huang*¶, Shih-Wei Liu*, I-Lung Chien, Yi-Hao Lin**, Miao-Ju Huang*****

** Department of Chemical Engineering, National Taiwan University, Taipei 106, Taiwan*
¶ (Tel: 886-2-23638999; e-mail: huanghpc@ntu.edu.tw)
*** Department of Chemical Engineering, National Taiwan University of Science and Technology, Taipei 106, Taiwan*
**** College of Medicine, Chang Gung University, Tao-Yuan, Taiwan*

Abstract: Type 1 diabetes mellitus must rely on daily insulin injection/infusion for the control of blood glucose. The treatments on those patients to maintain their blood glucose within an acceptable level is thus of essential importance. A good mathematical model of blood glucose may facilitate such a control. In this research, modeling and open-loop control of blood glucose for a type-1 patient using a model extended from the works of Hovorka and his coworkers (Hovorka *et al*., 2002; Hovorka *et al.*, 2004; Wilinska *et al*., 2005) are studied. Clinical data from a continuous glucose monitoring system is used to develop the model for such use. Open-loop control strategies for patients that use basal and bolus subcutaneous infusions via an insulin pump are presented.

1. INTRODUCTION

Some relevant studies indicate that good metabolic control of diabetes may decrease the risk of chronic complications. The averaged blood glucose (BG) levels as reflected by the HAb1c levels are generally considered an indication of goodness of such a control. Nevertheless, a good control in terms of HAb1c levels is only necessary but not sufficient, because, high glycemic excursion can not be detected by measuring the HAb1c values only. Recently, a system known as CGMS (Continuous Glucose Monitoring System) has been available for continuously measuring glucose concentrations in subcutaneous tissue. Clinic physicians can make uses of the recorded continuous profiles to learn the excursion of BG and modify their treatments on the patients. Meanwhile, the appearing of real time CGMS on the market also provides a basis for implementation of on-line blood glucose control in the future.

The motivation of this paper is to show, by making uses of a modification to the Harvoka's model (Hovorka *et al*., 2002; Hovorka *et al.*, 2004; Wilinska *et al*., 2005) and real CGMS data, an open-loop control that aims to desired HAb1c and blood glucose levels for type 1 patients can be developed.

2. MODEL DESCRIPTIONS

As mentioned, many physiological models have been proposed that describe glucose and/or insulin dynamics. In this paper, the model for study is based on the works of Hovorka and his coworkers (Hovorka, *et al*. ,2002; Hovorka, *et al*. ,2004; and Wilinska *et al*. ,2005). The reasons that this model is adopted for study is due to the inclusion of more detail insulin action that describes the physiological effect of insulin on glucose transport, removal and endogenous glucose production. Also it provides the insulin absorption through two compartmental channels that can be used to model the short acting and long acting effects from the bolus and basal insulin. Two subsystems are used to describe the glucose concentrations in the accessible compartment such as vein and organs, where measurements are made, and the inaccessible compartment such as tissue in human body, where measurements are not made. The insulin action describes the physiological effect of insulin on glucose transport, removal and endogenous glucose production. The original Harvokal model for IVGTT test is given as the following:

$$\frac{dQ_1}{dt} = -F_{01}^c - x_1(t)Q_1 + k_{12}Q_2(t) - F_R + U_G + \mathrm{W}\cdot EGP_0\left[1 - x_3(t)\right]; \quad (1)$$

$$\frac{dQ_2}{dt} = x_1(t)Q_1(t) - \left[k_{12} + x_2(t)\right]Q_2(t) \quad (2)$$

$$U_G = \sum_{i=1}^{N} \frac{D_G A_G (t - T_i) e^{\frac{-(t-T_i)}{t_{\max,G}}}}{t_{\max,G}^2} S(t - T_i); \quad T_i, i = 1, \cdots, M \quad (3)$$

$$F_R = \begin{cases} 0.003(G-9)V_G \cdot W & \text{, for G} \geq 9 mmole/L \\ 0 \text{ , else} \end{cases};$$

$$F_{01}^c = \begin{cases} F_{01} \cdot W & \text{; if } G \geq 4.5 mmole/L \\ F_{01} G \cdot W / 4.5 & \text{; otherwise} \end{cases};$$

and $G(t) = \dfrac{Q_1}{V_G W}$ $(mmole/L)$

$$\frac{dq_{1a}}{dt} = k_{ba}u_{ba} + k_{bo}\sum_i \frac{u_{bo,i}}{\tau}\exp\left\{-\frac{(t-T_i)}{\tau}\right\} - k_{a1^*}\cdot q_{1a} - \frac{V_{\max,LD}q_{1a}}{\left(k_{M,LD}+q_{1a}\right)} \quad (4)$$

$$\frac{dq_{1b}}{dt} = (1-k_{ba})u_{ba} + (1-k_{bo})\sum_i \frac{u_{bo,i}}{\tau}\exp\left\{-\frac{(t-T_i)}{\tau}\right\} - k_{a2^*}\cdot q_{1b} - \frac{V_{\max,LD}q_{1b}}{\left(k_{M,LD}+q_{1b}\right)} \quad (5)$$

$$\frac{dq_2}{dt} = k_{a1*}q_{1a} - k_{a1*}q_2 \quad (6)$$

$$\frac{dq_3}{dt} = k_{a1*}q_2 + k_{a2*}q_{1b} - k_e q_3 \quad (7)$$

$$\frac{dx_1}{dt} = -k_{a1}x_1(t) + \frac{k_{a1}S_{IT}^f q_3}{V_I\cdot W} \quad (8)$$

$$\frac{dx_2}{dt} = -k_{a2}x_2(t) + \frac{k_{a2}S_{ID}^f q_3}{V_I\cdot W} \quad (9)$$

$$\frac{dx_3}{dt} = -k_{a3}x_3(t) + \frac{k_{a3}S_{IE}^f q_3}{V_I\cdot W} \quad (10)$$

where, Q_1 and Q_2 (*mmole/l*) represent the glucose in accessible and non-accessible compartments, G (*mmole/l*) is the measurable glucose concentration, k_{12} represents the transfer rate constant from non-accessible to accessible compartment, V_G represents the distribution volume of the accessible compartment, and EGP_0 represents endogenous glucose production at the zero insulin concentration. F_{01}^c is the non-insulin-dependent glucose flux and F_R is the renal glucose clearance thresholds of 9 mmol/L. W is the patient's weight. U_G is the gut absorption rate, $t_{\max,G}$ is the time-of-maximum appearance rate of glucose in the accessible glucose compartment, D_G is the amount of carbohydrates (CHO) digested and A_G is the carbohydrate bioavailability. Variable q_2 represents the insulin mass (mU) in the nonaccessible subcutaneous compartment, q_3 represents the insulin mass (mU) in the plasma compartment. The quantities q_{1a} and q_{1b} represent the masses of insulin administered as continuous infusion (mU) through the slow and fast compartment channels (Wilinska *et al*. 2005). The variable u represents the basal insulin input (mU/min). The parameters k_{a1^*}, k_{a2^*}, and k_e are transfer rates ($\min^{-1}$), $V_{\max,LD}$ is the saturation level (mU/min) for Michaelis-Menten dynamics of insulin degradation, $k_{M,LD}$ is the value of insulin mass (mU) at which insulin degradation is equal to half of its maximal value for continuous infusion. The dimensionless constant k and $1-k$ represent the proportions of the total input flux of insulin passing through the slower and faster compartment channels, respectively. The variables, x_1, x_2, and x_3 represent the (remote) effects of insulin on glucose distribution/transport, glucose disposal and endogenous glucose production (Harvoka, *et al*. 2002). Finally, k_{ai}, i=1,⋯,3, represent the deactivation rate constants.

The modeling form given above is based on the work of Seborg and his coworkers (2008). On this basis, the addition of carbohydrates in-take from different meals (i.e. Eq.(3)) and the basal insulin and bolus insulin infusion/injection at different times (i.e. Eq.(4) and Eq.(5)) are considered as an extensions to the original model. Notations u_{ba} and u_{bo} are used to designate the quantities of basal insulin and bolus insulin, respectively.

The complete model for modelling the BG consists of equations from Eq.(1) through Eq.(10). V_G and A_G are assumed to be constant as: V_G=0.16 ($L^{-1}\cdot kg$) and A_G=0.8. Totally seventeen parameters in the model will be determined:

$$F_{01},\ k_{12},\ EGP_0,\ k_{a1^*},\ k_{a2^*},\ t_{MAX,G},\ S_{IT}^f/V_I,\ S_{ID}^f/V_I,\ S_{IE}^f/V_I,$$
$$V_{\max,LD},\ k_{M,LD},\ k_{ba},\ k_{bo},\ k_e,\ k_{a1},\ k_{a2},\ k_{a3}$$

For patient undertaking same insulin in basal and bolus injections, k_{ba} and k_{bo} in the model are assumed to have the same value.

3. PARAMETER ESTIMATION FOR MODELING

The model described above was applied to one real patient who has type-1 diabetes. The subject undergoing the experiment weighted 55kg and wore an insulin pump for insulin infusion. She also wore the MiniMed CGMS for five days during the experiment period. During that period, the meal contents and the insulin doses were recorded on a diary. These meal contents then were quantified by a dietician. No special arrangement was made for this experiment. The patient was asked to live on her normal way with meals and works as usual. The data from the patient accompanied with a complete diary on meals and insulin dosages are recorded. These data are then fitted into the model abovementioned. The modelling is aimed to find the parameters that minimize the sum of squares of the output errors. To compute these output errors, BG is computed by integrating the modified Hovorka model described in Section 2, starting with a set of parameters and initial conditions. The initial parameters are taken from the parameters of Marchetti *et al*. (2008). The steady-states values of the modified Hovorka model which correspond to a fasting level of BG are prepared in advance from the same model and are taken to initiate the integration for optimization. These initial states for integration are then updated from iteration to iteration using the resulted states in the last fasting stage of the previous run.

The modelling starts to fit themodel to the CGMS data of the first two days by making uses of the reported quantified CHOs and insulin doses. The parameters thus obtained are given as follows:

Parameter	Value	Unit	Parameter	Value	Unit
S_{IT}^f / V_I	0.019	$Min^{-1} \cdot mU^{-1} \cdot kg$	k_{12}	0.237	min^{-1}
S_{ID}^f / V_I	0.003	$Min^{-1} \cdot mU^{-1} \cdot kg$	k_{a1}	0.017	min^{-1}
S_{IE}^f / V_I	0.052	$mU^{-1} \cdot kg$	k_{a2}	0.273	min^{-1}
EGP_0	0.051	$mmole \cdot kg^{-1} \cdot min^{-1}$	k_{a3}	0.128	min^{-1}
F_{01}^c	1.899	$mmole \cdot min^{-1}$	k_e	0.036	min^{-1}
k	0.34	–	$t_{MAX,G}$	22.91	min
$V_{MAX,LD}$	2.156	$mU \cdot min^{-1}$	k_{a1*}	0.005	min^{-1}
$k_{M,LD}$	64.182	mU	k_{a2*}	0.054	min^{-1}

With the resulting parameters, the fitting of the real CGMS data to the model is shown in Figure 1. In this figure, the fitting of the model to the reported CGMS data in the first 48 hours looks good. The model is then used to predict the blood glucose in the remaining days. Figure 2 shows that the fitting is not good enough, but the trend is alright. The lack of fit in the extending time horizon is due to imprecise quantification of CHO intakes. If the remaining CHOs are allowed for some modifications, the fitting turns out to be more satisfactory (see Figure 3).

4. OPEN-LOOP CONTROL VIA SUBCUTANEOUS INSULIN INFUSION/INJECTION

Using an artificial pancreas, subcutaneous infusion of insulin according to a pre-programmed basal and bolus dosages is one approach to control the blood glucose of a type-1 diabetic patient in an open-loop manner. However, artificial pancreas is somewhat an expensive device that most of the patients may not afford. Fortunately, due to the availability of effective long acting insulin, the pre-programmed subcutaneous insulin injection can also be applied to patients who do not use artificial pancreas. The multiple subcutaneous injections with long acting and short acting insulin can be used to mimic the insulin secretions in a normal body. The basal insulin rate is aimed to maintain a given fasting level of blood glucose (e.g. 100 mg/dl). The bolus insulin dose is taken to enhance the control of blood glucose at each CHO intake.

4.1 Development of bolus dosage plan for Prandial CHO-uptake

While planning the bolus dosage under a prandial condition, some constraints should be considered. These constraints come from clinical demands for normal control of blood glucose. For example,

(1). Fasting blood glucose ≤ 100 mg/dl (≤ 5.6 mmol/l)

(2). Two-hour Post-prandial blood glucose $\leq$ 120 mg/dl

(3). In all time, blood glucose $\geq$ 70 mg/dl and never less than 50 mg/dl

The above standards may not be achievable by a type-1 diabetic patient in real practice. Apart from keeping patients from a hyperglycemia status, type-1 diabetic patient should be cautious to keep from having hypoglycaemia, especially during midnight (i.e. approx. 6 hours after dinner). Thus, in order to accommodate properly the blood glucose concerns in many aspects, the determination of bolus dosage for a CHO-intake is thus formulated as the problem of the following:

$$\underset{u_{bo}}{Min}\left\{\left|G_{av}(t^*, u_{bo}(x_i)) - \gamma_1^*\right| \text{ given } | \, CHO = x_i \text{ and } G_{fasting} = \gamma_o\right\} \quad (13)$$

$$s.t. \quad \begin{cases} (1)\ G(t) \geq \gamma_*, \\ (2)\ G(t) < \gamma^* \ \forall t \geq 0\ , \end{cases}$$

Where, x_i, i=1, 2, 3 is the grams of CHO-intake in each meal, $\gamma^*, and\ \gamma_*$ are parameters to be assigned, which may vary according to the physician on a patient-to-patient basis for constraining blood glucose. Notice that, here, γ_* is taken as 70mg/dl. In other words, $\underline{G} \geq 70\ mg/dl$ is a hard constraint for patient to avoid from having hypoglycemia. The value of t^* is taken as 6hrs, since each meal is usually 6 hours apart from one to the other. $G_{av}(6hrs, u_{bo}(x_i))$ is the average value of G in a 6-hours period with x_i CHO-intake and $u_{bo}(x_i)$ bolus dosage.

To solve the problem in Eq. (13), first, we need the function $u_{bo}(x_i)$ and a basal insulin amount, $u_{ba}(G_{fasting})$, to maintain the blood glucose at a specified fasting level. The latter is used to mimic the secretion of the basal insulin in a normal human body. Mathematically, basal insulin can be considered as a step dose lasting for 24 hours a day that leads G to a fasting level. In fact, this value can be obtained by solving the set of algebraic equations obtained from setting the derivatives in the extended Hovorka model to zeros. The fasting glucose to mimic the effect of the secretion of basal insulin in a normal human body is taken as 100 mg/dl in this study (γ_0=100 mg/dl). A lower value may also be taken, however, a too low value may lead Eq.(13) without having feasible solution. Based upon this basal insulin amount (or, in other words, the targeting blood glucose) the bolus dosage plan will then be computed.

Next, we need to develop the function, $u_{bo}(x_i)$, which describes the required bolus dosage to a CHO intake from meals. The parameters $\gamma^*, and\ \gamma_1^*$ are considered for type-1 diabetic patients to have an acceptable glycated hemoglobin (i.e., hemoglobin HbA_{1c} or, simply, A1c) value. The use of hemoglobin A1c for monitoring the degree of control of glucose metabolism in diabetic patients was proposed in 1976 by Koenig and coworkers. A buildup of A1c within the red cell reflects the average level of glucose to which the cell has been exposed during its life cycle (approx. 120 days). Thus,

the A1c level is proportional to the averaged blood glucose concentration over the previous four weeks to three months. According to ADA (American Diabetes Association), the mean plasma glucose concentration (MPG) is related to HbA_{1c} with an empirical equation of the following:

MPG (mg/dl) = (35.6* HbA_{1c}) - 77.3, or, MPG (mmol/l) = (1.98* HbA_{1c}) - 4.29 (14)

A simpler and approximately equivalent formula can also be written as:

$$MPG(mg/dl) = 35*(A1c-5)+100 \quad (15)$$

Where, MPG is the mean plasma glucose concentration in the last three months. Thus,

$$A1c = \left(MPG-100\right) \div 35+5 \quad (16)$$

If the daily average blood glucoses (i.e. $G_{av}(24hrs)$) are equal from day to day in a period of at least three months, the MPG will equal to its daily time-average. Notice that the glucose concentration in the plasma is higher than the glucose concentration in the whole blood by about 11%. In the mathematical model, the glucose concentration G is referred to the whole blood. As a result, in terms of blood glucose as described by the model, Eq.(16) needs to be updated as:

$$A1c \cong \left[G_{av}(24hrs)\times 1.1-100\right] \div 35+5 \quad (17)$$

For easier application, one may consider to replace the $G_{av}(24hrs)$ with the following:

$$G_{av}(24hrs) \cong \left[\sum_{i=1}^{3} G_{av}(6hrs, u_{bo}(x_i)) + G_{fasting}\right] * 0.25 \quad (18)$$

And,

$$A1c \cong \left[\left\{\sum_{i=1}^{3} G_{av}(6hrs, u_{bo}(x_i)) + G_{fasting}\right\} * 0.25\times 1.1-100\right] \div 35+5 \quad (19)$$

Where, x_1, x_2, x_3 are CHO amounts in terms of grams taken from the daily three meals. Eq.(18) compute the mean BG by assuming that daily BG has a level at $G_{av}(6hrs)$ in a period of three meals (i.e.18 hours) and at 100mg/dl in the fasting period (i.e. another 6 hrs). Thus, specifying γ_1^* in Eq.(13) is to specify the target A1c as shown in Figure 4. An effective control of blood glucose should make this A1c value to remain below 7. In practical treatment, the values of γ^*, and γ_1^* may differ from subject to subject. If a bolus-to-CHO relation is available, then, given a daily CHO-plan, the average blood glucose can be computed from integrating the model. A graphical approach to the solution of a bolus-to-CHO relation will be demonstrated in the example that follows.

4.2 Case study

As mentioned, the basal insulin amount versus the fasting blood glucose levels can be calculated from the steady-state solution to the modified Hovorka model given in Section 2. For this case, the basal insulin amounts corresponding to each possible fasting glucose level are prepared and plotted in Figures 4. On the other hand, under a basal dose that leads to a fasting glucose of 100mg/dl, the responses of the blood glucose to different CHO intakes are computed. Based on these responses, the average blood glucoses at 4hrs and 6 hrs, together with maximum and minimum values under the same dosage are computed and plotted in Figure 5 and Figure 6, , with an insulin increment of 0.5 unit. The values of γ^*, and γ_1^* are then specified in order to obtain the bolus dose. As an example in this case, they are selected as 320, and 120, respectively.

The feasible regions to satisfy condition (1) and condition (2) of Eq. (13) are plotted for specific CHO intakes, starting from 15grams to 60grams, with an increment of 15grams. As shown in Figure 5, the feasible region that satisfies condition (1) and condition (2) is the region spanned by the pink and red lines. In each case (e.g. Figure 5 and Figure 6), there is u_{bo} that corresponding to the given CHO value, x. However, in case there is no feasible solution exists that will give $G_{av}(6hrs)$ exactly at 120 mg/dl, the bolus dosage at the boundary of the feasible region will be taken. By increasing the CHO amount, x, at an increment of 15 grams, a curve of bolus insulin required for different prandial CHOs can be obtained as shown in Figure 7. Thus, under a give basal insulin amount, the bolus dosage required for each CHO-intake can be read from these curves. If, a CHO meal-plan is taken (e.g., breakfast: 30gm, lunch: 60gm, dinner:60gm) with the bolus dosage plan as shown in Fig. 7 (e.g. breakfast: 1.05U, lunch: 2.1U, dinner: 2.1U), the averaged blood glucose could be expected to be 113.5 which gives 5.76 for the A1c value. Compared to the current record from the CGMS, the A1c value as well as the blood glucose could be significantly improved.

5. CONCLUSIONS

Modeling glucose-insulin interactions with real CGMS data and a model extended from Hovorka and his coworkers (Hovorka *et al*., 2002; Hovorka *et al.*, 2004) are studied. Using the mathematical model as a frame work, dynamic models are built for a case study. Data from a continuous glucose monitoring system (CGMS, MiniMed) are used to estimate the parameters in the model. As the CHO contents in various foods are fuzzy due to natural language, the CHO values thus quantified are subject to certain extent of uncertainties. As a result, during the parameter estimation, these quantified CHO values need to be modified. The resulting models are then used to determine the daily basal and bolus dosages to mimic the secretion of insulin of a human body and control the blood glucose to an acceptable level. The basal amount is to be determined to maintain the fasting glucose value at a given level. The bolus dosage is then determined based upon this basal insulin and the CHO intake in each meal. The bolus is aimed to keep the blood glucose within upper and lower bounds. The lower bound is

70 mg/dl which is normally considered in medical treatment to prevent the occurrence of hypoglycermia. The upper bound can be considered on a person-to-person basis. Besides setting the glucose value in the upper/lower bounds, the bolus dosage plan is also aimed to have a targeted A1c value, which is normally less than 7.0 in general medical treatments for the diabetes patients. The basal insulin amount and the bolus dosage plan are demonstrated with the utilization of the developed models.

REFERENCES

1. Hovorka, R. F. Shojaee-Moradie, P. V. Carroll, L. J. Chassin, I. J. Gowrie, N. C. Jackson, R. S. Tudor, A. M. Umpleby, and R. H. Jones, "Partitioning glucose distribution/transport, disposal, and endogenous production during IVGTT," *Am J Physiol Endocrinol Metab*, vol. 282, no. 5, pp. E992–1007, May 2002.

2. Hovorka R., V. Canonico, L. J. Chassin, U. Haueter, M. Massi-Benedetti, M. O. Federici, T. R Pieber, H. C Schaller, L. Schaupp, T. Vering and M. EWilinska", Nonlinear model predictive control of glucose concentration in subjects with type 1 diabetes" Physiol. Meas. vol.25(2004), pp.905–920.

3. Koenig RJ, Peterson CM, Jones RL, Saudek C, Lehrman M, Cerami A (1976). "Correlation of glucose regulation and hemoglobin AIc in diabetes mellitus". *N. Engl. J. Med.* **295** (8): 417–20.

4. Gianni Marchetti , M. Barolo , L. Jovanovic , H. Zisser , D. E. Seborg, "A feedforward-feedback glucose control strategy for type 1 diabetes mellitus", JPC, 18 (2008) 149-162.

5. Wilinska M. E., L. J. Chassin, H. C. Schaller, L. Schaupp, T. R. Pieber, and R. Hovorka, "Insulin kinetics in type–1 diabetes: Continuous and bolus delivery of rapid acting insulin," IEEE Trans. Biomed. Eng., vol. 52, no. 1, pp. 3–12, 2005.

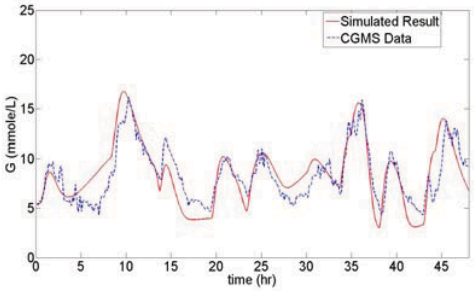


Figure 1. Fitting of the CGMS data to the model

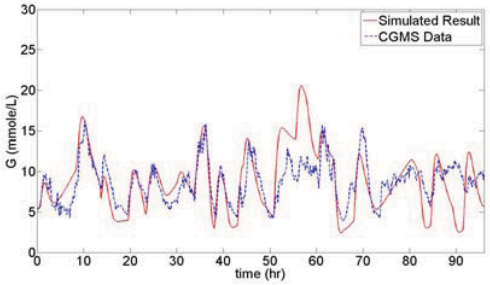


Figure 2. Validation of model using the original CHO and insulin data (case 1)

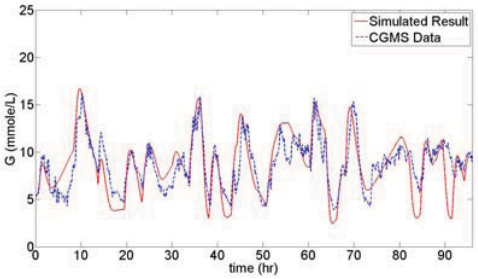


Figure 3. The blood glucose excursions after meal CHO being modified (Case 1).

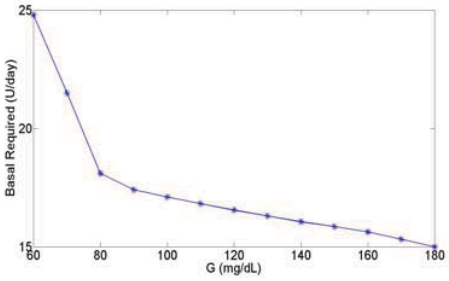


Figure 4. The basal dose for fasting blood glucose

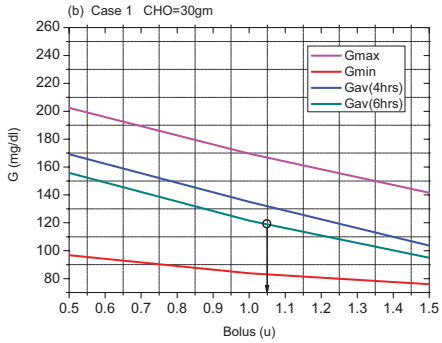


Figure 5. Graphical solution to Eq.(13), CHO intakes at 30gms.

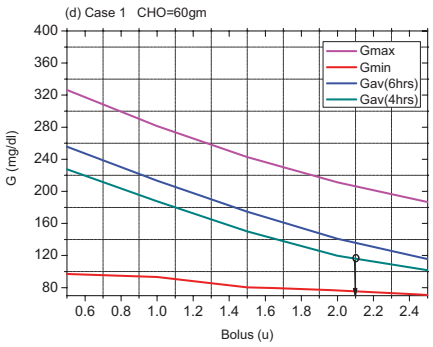


Figure 6. Graphical solution to Eq.(13), CHO intakes at 60gms.

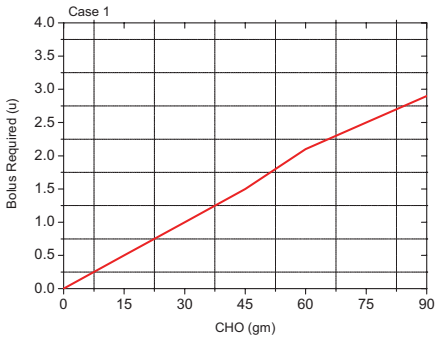


Figure 7. Bolus dose v.s. CHO for Case 1 ($G_{fasting}$=100mg/dl)

Table 4. CHO-plans vs A1c for Case 1

A1c*: computed from G_{av}(6hrs)

CHO-plan (g)			MG (24hr)	MPG (24hr)	A1c	A1c*
B	L	D	mg/dL			
15	45	45	114.52	125.97	5.7	5.7
15	45	60	113.94	125.33	5.7	5.7
15	60	45	111.58	122.74	5.6	5.7
15	60	60	111.22	122.35	5.6	5.7
30	45	45	112.39	123.62	5.7	5.7
30	45	60	111.94	123.14	5.7	5.7
30	60	45	110.63	121.70	5.6	5.7
30	60	60	110.35	121.39	5.6	5.7

Nonlinear Model-Based Control of an Experimental Reverse Osmosis Water Desalination System ⋆

Alex R. Bartman, [*] **Panagiotis D. Christofides,** [*,**,1] **Yoram Cohen** [*]

[*] *Department of Chemical and Biomolecular Engineering, University of California, Los Angeles, CA 90095-1592 USA.*
[**] *Department of Electrical Engineering, University of California, Los Angeles, CA 90095-1592, USA.*

Abstract: This work focuses on the design and implementation of a nonlinear model-based control system on an experimental reverse osmosis (RO) membrane water desalination system in order to deal with large set-point changes and variations in feed water salinity. A dynamic nonlinear lumped-parameter model is derived using first-principles and its parameters are computed from experimental data to minimize the error between model predictions and experimental RO system response. Then, this model is used as the basis for the design of a nonlinear control system using geometric control techniques. The nonlinear control system is implemented on the experimental RO system and its set-point tracking capabilities are successfully evaluated.

Keywords: Process control, process monitoring, model based control, nonlinear process systems

1. INTRODUCTION

Reverse osmosis (RO) membrane desalination has emerged as one of the leading methods for water desalination due to the low cost and energy efficiency of the process (Rahardianto et al. (2007)). Lack of fresh water sources has necessitated further development of these desalination plants, especially in areas with dry climates. Even with advances in reverse osmosis membrane technology, maintaining the desired process conditions is essential to successfully operating a reverse osmosis desalination system. Seasonal, monthly, or even daily changes in feed water quality can drastically alter the conditions in the reverse osmosis membrane modules, leading to decreased water production, sub-optimal system performance, or even permanent membrane damage. In order to account for the variability of feed water quality, a robust process control strategy is necessary. In a modern reverse osmosis (RO) plant, automation and reliability are elements crucial to personnel safety, product water quality, meeting environmental constraints, and satisfying economic demands. Industrial reverse osmosis desalination processes primarily use traditional proportional and proportional-integral control to monitor production flow and adjust feed pumps accordingly (Alatiqi et al. (1999)). While such control strategies are able to maintain a consistent product water (permeate) flow rate, they may fail to provide an optimal closed-loop response with respect to set-point transitions owing to the presence of nonlinear process behavior (Chen et al. (2005)). In some cases, permeate production can decrease due to scaling or fouling on the membrane surface. When this occurs, traditional control algorithms force the feed pumps to increase feed flow rate leading to an increased rate of scaling, irreversible membrane damage, and eventual plant shutdown. Traditional process control schemes are also unable to monitor plant energy usage and make adjustments toward energy-optimal operation. Model based control is a promising alternative to traditional RO plant control strategies. Several model based methods such as model-predictive control (MPC) and Lyapunov-based control have been evaluated via computer simulations for use in reverse osmosis desalination (Abbas (2006); McFall et al. (2008); Bartman et al. (2009b); Gambier and Badreddin (2002)). Experimental system identification and MPC applications can also be found in the literature (Assef et al. (1997); Burden et al. (2001)). Model based control methods have also been used in conjunction with fault detection and isolation schemes to improve robustness of control methods in the presence of sensor and actuator failures (McFall et al. (2008)). Other automatic control methods utilize model based control based on a linear model (Alatiqi et al. (1989)); using step tests to create a model that is a linear approximation around the desired operating point. Several other traditional control methods have also been studied in the context of RO system integration with renewable energy sources (Herold and Neskakis (2001); Liu et al. (2002)). Motivated by these considerations, the goal of this work is to evaluate the effectiveness of a feedback linearizing nonlinear model-based controller through application to an experimental reverse osmosis desalination system. The nonlinear model-based controller is shown to possess excellent set-point tracking capabilities. The nonlinear controller is also shown to outperform a proportional-integral control system.

⋆ Financial support from NSF and the California Department of Water Resources is gratefully acknowledged.
[1] Corresponding author: P.D. Christofides, pdc@seas.ucla.edu

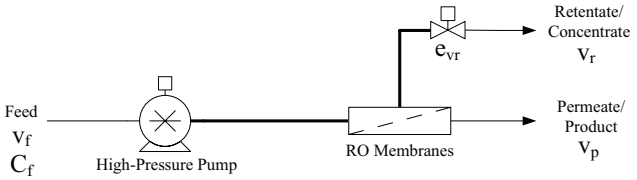


Fig. 1. Reverse osmosis system used for model development

2. RO SYSTEM MODEL

In this section, a fundamental model of a representative RO desalination system is developed including all of the basic elements present in UCLA's experimental RO desalination system. In this system, shown in Fig. 1, water enters the feed pump, which is equipped with a variable frequency drive (VFD), and is pressurized to the feed pressure P_{sys}. The pressurized stream enters the membrane module where it is separated into a low-salinity product (or permeate) stream with velocity v_p, and a high-salinity brine (or retentate) stream with velocity v_r. In the model, the individual spiral-wound membranes in series are assumed to be one large spiral-wound membrane in one large vessel, where any effects of individual membrane vessel interconnections are neglected. The pressure downstream of the actuated valve and at the permeate outlet is assumed to be equal to atmospheric pressure.

The model is based on a mass balance taken around the entire system and an energy balance taken around the actuated retentate valve. In the model derivation, it is assumed that the water is an incompressible fluid, all components are operated on the same plane (so potential energy terms due to gravity can be neglected), and the density of the water is assumed to be constant. It is also assumed that the effective concentration in the membrane module is a weighted average of the feed concentration and the brine stream concentration (see Eq. 6 below). The model derivation results in a nonlinear ordinary differential equation for the retentate stream velocity and an algebraic relation for the system pressure. This model is an adaptation of a model developed in our previous work used to describe a similar reverse osmosis desalination system (McFall et al. (2008)). In the previous work, the system utilized a feed pump with a constant feed flow rate, but used a separate bypass stream with an actuated valve to control the velocity of the water feeding to the membrane units. An equation for the osmotic pressure based on effective concentration and temperature in the membrane unit was also developed in (Lu et al. (2007)), and is used as an estimate in the model. Specifically, an energy balance is first taken around the retentate valve which leads to the following differential equation:

$$\frac{dv_r}{dt} = \frac{P_{sys}A_p}{\rho V} - \frac{1}{2}\frac{A_p e_{vr} v_r^2}{V} \tag{1}$$

where v_r is the retentate stream velocity, P_{sys} is the system pressure, A_p is the pipe cross-sectional area, ρ is the fluid density, V is the system volume and e_{vr} is the retentate valve resistance. To compute an expression for the system pressure in terms of the other process variables, an overall steady-state mass balance is taken to yield:

$$0 = v_f - v_r - v_p \tag{2}$$

where v_f is the feed stream velocity and v_p is the permeate stream velocity. In order to get an expression for the system pressure, the following classical expression is used for the computation of the permeate stream velocity:

$$v_p = \frac{A_m K_m}{\rho A_p}(P_{sys} - \Delta\pi) \tag{3}$$

where A_m is the membrane area, K_m is the membrane overall mass transfer coefficient, and $\Delta\pi$ is the difference in osmotic pressure between the feed side of the membrane and the permeate side. Substituting Eq. 3 into Eq. 2, the following expression for the system pressure (P_{sys}) is obtained:

$$P_{sys} = \frac{\rho A_p}{A_m K_m}(v_f - v_r) + \Delta\pi \tag{4}$$

where the osmotic pressure ($\Delta\pi$) and effective average concentration at the membrane surface (C_{eff}) on the feed side can be computed from the following relations:

$$\Delta\pi = \delta C_{eff}(T + 273) \tag{5}$$

$$C_{eff} = C_f(a + (1-a)((1-R) + R(\frac{v_f}{v_r}))) \tag{6}$$

where C_f is the amount of total dissolved solids (TDS) in the feed, a is an effective concentration weighting coefficient, δ is a constant relating effective concentration to osmotic pressure, T is the water temperature in degrees Celsius, and R is the fractional salt rejection of the membrane. Substituting Eq. 4 into the energy balance equation of Eq. 1 yields the following nonlinear ordinary differential equation for the dynamics of the retentate stream velocity:

$$\frac{dv_r}{dt} = \frac{A_p^2}{A_m K_m V}(v_f - v_r) + \frac{A_p}{\rho V}\Delta\pi - \frac{1}{2}\frac{A_p e_{vr} v_r^2}{V} \tag{7}$$

Using the above dynamic equation, various control techniques can be applied using the valve resistance value (e_{vr}) as the manipulated input. As the valve resistance goes to zero, the valve behaves as an open pipe; as the valve resistance approaches infinity, the valve behaves as a total obstruction and the flow velocity goes to zero (Bird et al. (2002)). To accurately model the valve dynamics and to relate the experimental results to the concept of valve resistance value (e_{vr}), the concept of valve C_v is used. The definition of C_v for a valve in a water system is:

$$C_v = \frac{Q_r}{\sqrt{P_{sys}}} \tag{8}$$

where Q_r is the volumetric flow rate ($Q_r = A_p v_r$) through the retentate valve. In order to obtain an expression for C_v as a function of the retentate valve resistance (e_{vr}), we consider the steady state form of the energy balance of Eq. 1, solve the resulting equation for P_{sys} and substitute the resulting expression for P_{sys} into Eq. 8 to yield:

$$C_v = \frac{A_p}{\sqrt{\frac{1}{2}\rho e_{vr}}} \tag{9}$$

Depending on the type of valve and its flow characteristics, it is assumed that the C_v values (and in turn, the e_{vr}

values) can be related to the valve position (percentage open) through the following empirical logarithmic relation based on commercially available valve data (Bartman et al. (2009b)):

$$O_p = \mu \ln e_{vr} + \phi \tag{10}$$

where μ and ϕ are constants depending on the valve properties. The values of μ and ϕ for this model are taken from a paper based on the same experimental system at UCLA (Bartman et al. (2009b)). For the model presented in this paper, the curve relating valve position (O_p) to resistance value (e_{vr}) is shown in Fig. 2. It can be seen in Fig. 2 that as the valve position goes to zero (fully closed), the valve resistance values begin to grow at an increasing rate; and as the valve approaches the fully-open position, the resistance values change slowly. The data from the experimental system is also plotted on the figure, and it can be seen that the data does not fit the same logarithmic relation as the ideal valve curve. Due to the shape of the experimental data curve, the data is fit in three segments with curve fits following a similar form as the theoretical curve. The first curve fit is applied to valve resistance (e_{vr}) values of approximately 205 to 212 and takes the form:

$$O_p = -84.428 \ln(e_{vr}) + 459.21 \tag{11}$$

For e_{vr} values between 212 and 6200, O_p is computed by:

$$O_p = -2.0473 \ln(e_{vr}) + 18.141 \tag{12}$$

while for e_{vr} values above 6200, O_p is computed by:

$$O_p = -0.0778 \ln(e_{vr}) + 0.9476 \tag{13}$$

This treatment of the valve characteristics allows for conversion of the experimental values of O_p to values of e_{vr} in the model-based nonlinear control algorithm, and allows for values of e_{vr} generated by the control algorithm to be translated to values of O_p to be sent to the actuated valve on the experimental system. Capturing the nonlinearity present in the valve is extremely crucial when applying the control algorithms to the experimental system.

2.1 Computation of Nonlinear Model Parameters Based on Experimental Data

Most of the parameters of the model of Eqs. 7-13 such as the membrane area (A_m), water density (ρ), pipe cross-sectional area (A_p), and system volume (V) have constant values which can be obtained from the experimental system. Another key model parameter, the overall mass transfer coefficient (K_m) was computed to match the model response to experimental step-test data. Specifically, K_m was computed using steady state data from the experimental system by minimizing the difference between the model steady state and the experimental system steady state for various step tests. The computed values of K_m were then averaged to determine the best value for use in the model used for controller design. The values of the model parameters can be found in Table 1.

3. CONTROL ALGORITHMS

Two separate control loops are present in the control problem formulation. The first loop regulates the system pressure by adjusting the variable frequency drive (VFD) speed directly (effectively changing the feed flow rate). This control loop will be termed "loop I". In each set of experiments presented below, a proportional-integral (PI) feedback controller is used to keep the system pressure (P_{sys}) at the set-point value (P_{sys}^{sp}) of 150 psi. This control algorithm takes the form:

$$S_{VFD} = K_f(P_{sys}^{sp} - P_{sys}) + \frac{K_f}{\tau_f}\int_0^{t_c}(P_{sys}^{sp} - P_{sys})dt \tag{14}$$

where S_{VFD} is the control action applied to the variable frequency drives (VFD speed), K_f is the proportional gain and τ_f is the integral time constant. The second control loop (termed "loop II") uses a nonlinear model-based controller (for the purposes of comparison, and a PI controller is also used in loop II). The nonlinear controller utilizes the error between the retentate velocity and its corresponding set-point, but it also takes into account many additional system variables (El-Farra and Christofides (2001, 2003); Christofides and El-Farra (2005)). Specifically, the nonlinear model-based controller manipulates the actuated retentate valve position by using measurements of the feed flow velocity (v_f), feed salinity (C_f), and retentate flow velocity (v_r). The nonlinear controller is designed following a feedback linearization approach. To derive the controller formula, the following linear, first-order response in the closed-loop system between v_r and v_r^{sp} is requested:

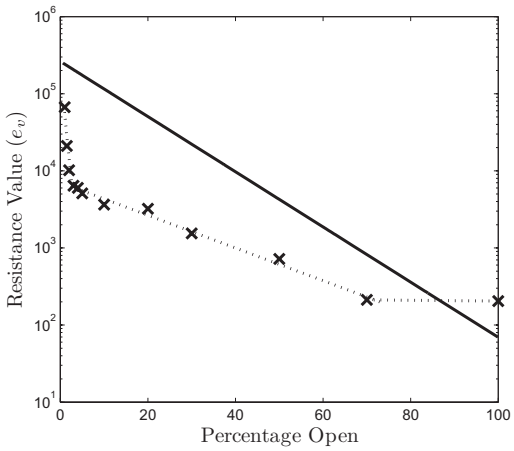


Fig. 2. Correlation between valve resistance value (e_{vr}) and valve percentage open (O_p) : commercial theoretical data (solid line), experimentally measured data (x), and curve fittings to experimental data (dashed lines) using Eqs. 11-13.

Table 1. Process model parameters based on experimental system data.

ρ	=	1007	kg/m^3
V	=	0.6	m^3
A_p	=	0.000127	m^2
A_m	=	15.6	m^2
K_m	=	6.4×10^{-9}	s/m
C_f	=	4842	mg/L
a	=	0.5	
T	=	22	$^\circ C$
R	=	0.97	

$$\frac{dv_r}{dt} = \frac{1}{\gamma}(v_r^{sp} - v_r) \tag{15}$$

It is noted that a first-order response is requested because the relative degree between v_r and e_{vr} is one (Christofides and El-Farra (2005)). Using this approach, the following formula is obtained for the nonlinear controller:

$$e_{vr} = \frac{\frac{1}{\gamma}(v_r^{sp} - v_r) - \frac{A_p^2}{A_m K_m V}(v_f - v_r) - \frac{A_p \delta (T+273)}{\rho V} C_{eff}}{\frac{-A_p}{2V}(v_r^2)} \tag{16}$$

To achieve offset-less response, integral action is added to the controller in Eq. 16 and the resulting controller takes the form:

$$e_{vr} = \frac{\frac{1}{\gamma}(v_r^{sp} - v_r) + \frac{1}{\tau_{NL}}\int_0^{t_c}(v_r^{sp} - v_r)dt}{\frac{-A_p}{2V}(v_r^2)} + \frac{-\frac{A_p^2}{A_m K_m V}(v_f - v_r) - \frac{A_p \delta (T+273)}{\rho V} C_{eff}}{\frac{-A_p}{2V}(v_r^2)} \tag{17}$$

As a baseline, the performance of the nonlinear controller is compared to a traditional form of control. Loop II, using PI control, uses the retentate (or concentrate) stream flow velocity to manipulate the actuated valve in order to regulate the retentate stream velocity/flow rate. Under PI control, the control system for loop II takes the form:

$$O_p = K_r(Q_r^{sp} - Q_r) + \frac{K_r}{\tau_r}\int_0^{t_c}(Q_r^{sp} - Q_r)dt \tag{18}$$

where Q_r is the retentate stream volumetric flow rate and Q_r^{sp} is the retentate stream flow rate set-point. In the experiments, the performance of the nonlinear controller implemented on the experimental system is compared to the performance of the nonlinear controller implemented on the process model and to the performance of the proportional-integral controller implemented on the experimental system. The control algorithms were programmed into the data acquisition and control software to operate in real-time with a sampling time of 0.1 seconds. Additionally, the actuated retentate valve is powered by an electric motor with a maximum operating speed which must be taken into account when attempting to simulate the nonlinear controller action. From testing on the experimental system, it was found that the actuated valve could travel its entire range in approximately 45 seconds; this provides an important constraint on the speed of valve opening/closing in the simulations of the form:

$$\left|\frac{dO_p}{dt}\right| \leq 2.22\frac{\%}{s} \tag{19}$$

To derive the constraint of Eq. 19, it is assumed that the valve speed is independent of valve position (valve always turns at maximum speed). This is a physical constraint which is intrinsically accounted for in the experimental results and is programmed into the nonlinear model-based controller simulation as well (to facilitate comparison). Additionally, when using the experimental system, the valve position is not allowed to fall under 1%, and any values sent to the valve above 100% are translated to the max value of 100% open. The lower constraint ($< 1\%$) is enforced so that the system pressure will not rise too rapidly. A constraint on the variable frequency drive is also placed to avoid pressure spikes (a maximum VFD speed of 4.5/10 is used). In the experiments presented in this work, the actuators do not reach these constraints.

4. EXPERIMENTAL SYSTEM DESCRIPTION

The experimental reverse osmosis water desalination system constructed at UCLA's Water Technology Research (WaTeR) Center was used for conducting the control experiments. This experimental system is comprised of a feed tank, two low-pressure feed pumps in parallel which provide enough pressure to pass the feed water through a series of cartridge filters while also providing sufficient pressure for operation of the high-pressure pumps, two high-pressure pumps in parallel (each capable of delivering approximately 4.3 gallons per minute at 1000 psi), and a bank of 18 pressure vessels containing Filmtec spiral-wound RO membranes. The high-pressure pumps are outfitted with variable frequency (or variable speed) drives which enable the control system to adjust the feed flow rate by using a 0-10V output signal. The bank of 18 membranes are arranged into 3 sets of 6 membranes in series; and for the control experiments presented below, only one bank of 6 membrane units was used. The experimental system uses solenoid valves controlled by the data acquisition and control hardware to enable switching between multiple arrangements of the membrane modules (2 banks of 6 in parallel to one bank of 6 in series, or any number of the modules in series) while also allowing for control of the flow direction through the membrane banks. After the membrane banks, an actuated valve is present to control the cross-flow velocity (v_r) in the membrane units, while also influencing system pressure. This valve is used as an actuator for the control system utilizing the control algorithms presented in section 3. The resulting permeate and retentate streams are currently fed back to the tank in an overall recycle mode, but for field operation the system can be operated in a one-pass fashion. The experimental system also has an extensive sensor and data acquisition network; flow rates and stream conductivities are available in real-time for the feed stream, retentate stream and permeate stream. The pressures before each high pressure pump, as well as the pressures before and after the membrane units (feed pressure and retentate pressure) are also measured. The system also includes sensors for measuring feed pH, permeate pH, in-tank turbidity, and feed turbidity after filtration (in real-time). A centralized data acquisition system takes all of the sensor outputs (0-5V, 0-10V, 4-20mA) and converts them to process variable values on the local (and web-accessible) user interface where the control system is implemented. The data is logged on a local computer as well as on a network database where the data can be accessed via the internet, while the control portion of the web-based user interface is only available to persons with proper authorization. The data acquisition and control system uses National Instruments software and hardware to collect the data at a sampling rate of 10 Hz and perform the necessary control calculations needed for the computation of the control action to be implemented by the control actuators. A photograph of the system can be seen in Fig. 3.

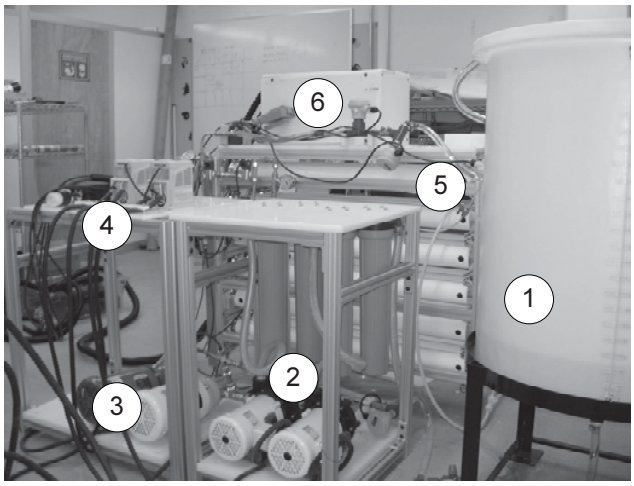


Fig. 3. UCLA experimental RO membrane water desalination system: (1) Feed tank, (2) Low-pressure pumps and prefiltration, (3) High-pressure positive displacement pumps, (4) Variable frequency drives (VFDs), (5) Pressure vessels containing spiral-wound membrane units (3 sets of 6 membranes in series), (6) National Instruments data acquisition hardware and various sensors.

5. EXPERIMENTAL CLOSED-LOOP RESULTS

In the control experiments presented in this paper, the experimental system was turned on and the PI loop controlling the variable frequency drives (loop I) was activated to bring the system pressure to a set-point of $P_{sys} = 150$ psi. The retentate flow rate was set to 1.5 gallons per minute (gpm). After the system had been operating at this steady state for a sufficient period of time, loop II was activated to manipulate the retentate valve. All data taken from the experimental system was averaged (after the experiments) using a 19 point moving average to remove most of the measurement noise. The following sets of experiments compare the performance of the nonlinear controller with the performance of the proportional and proportional-integral controllers. The closed-loop response observed for the nonlinear controller applied to the dynamic process model is used as a baseline for comparison of controller performance, as well as to determine an approximate range of controller tunings for the experimental system. In this set of experiments, the retentate flow rate set-point was changed from an initial value of 1.5 gpm to a new value of 0.8 gpm, while the VFD control loop is again maintained at a pressure set-point of 150 psi. In this set of experiments, the performance of the nonlinear controller with integral term is evaluated against the performance of a proportional-integral (PI) controller (both of these controllers are implemented experimentally), and the performance of the nonlinear controller with integral action applied to the dynamic process model via simulations. The feed salt concentration for these experiments was approximately 8200 ppm of NaCl. The tuning parameters for the controllers in this set of experiments can be found in Tables 2 and 3.

The results for these experiments are plotted in Figs. 4 - 5. In Fig. 4, it can be seen that all of the closed-loop results (simulated and experimental) decrease at the same rate initially (due to the valve opening/closing rate constraint). As expected, the simulated nonlinear model-based controller with integral term immediately converges to the set-point with no offset since it is not subject to any plant-model mismatch or measurement noise. As it is evident in Table 2, the integral time constant for the simulated controller is slightly different ($\tau_f = 0.01$, $\tau_f^{sim} = 0.0091$). The simulations where the nonlinear controller was applied to the process model were used to find an approximate range of controller parameters, but these values were implemented on the experimental system and changed slightly to achieve better closed-loop performance in the presence of plant-model mismatch. The speed of the closed-loop response under the nonlinear controller applied to the experimental system is slower in terms of convergence to the set-point than the one in the simulated case and the retentate flow rate reaches the set-point in about 145 seconds. The proportional-integral

Table 2. Loop I PI controller tuning parameters.

K_f	=	0.01
τ_f	=	0.1
K_f^{sim}	=	0.0091
τ_f^{sim}	=	0.1

Table 3. Loop II controller tuning parameters (both PI and nonlinear controllers).

K_r	=	1
τ_r	=	5
γ	=	0.6
τ_{NL}	=	10
γ^{sim}	=	0.6
τ_{NL}^{sim}	=	10

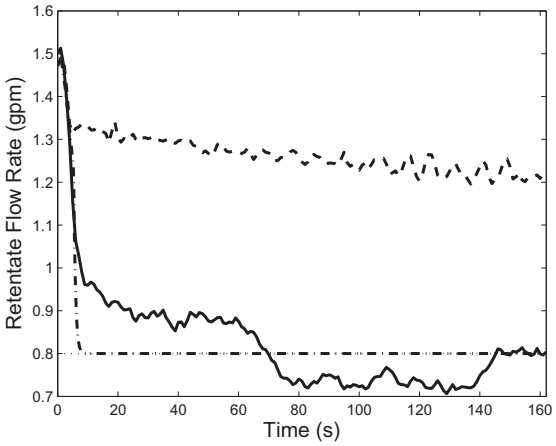


Fig. 4. Profiles of retentate flow rate (Q_r) with respect to time for retentate flow rate set-point transition from 1.5 to 0.8 gpm under proportional-integral control (dashed line), nonlinear model-based control with integral action (solid line) and nonlinear model-based control with integral action implemented via simulation on the process model (dash-dotted line). The horizontal dotted line denotes the retentate flow rate set-point ($Q_r^{sp} = 0.8$ gpm).

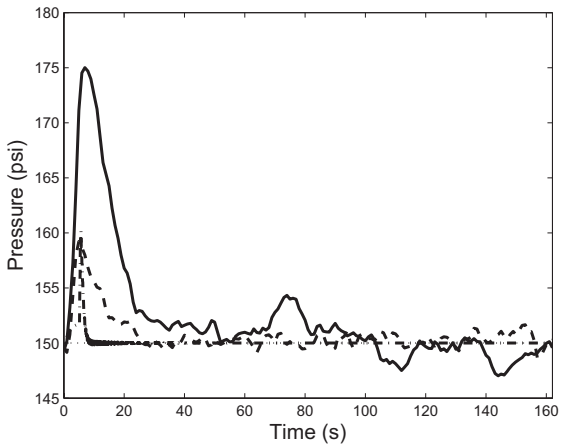


Fig. 5. Profiles of system pressure (P_{sys}) with respect to time for retentate flow rate set-point transition from 1.5 to 0.8 gpm under proportional-integral control (dashed line), nonlinear model-based control with integral action (solid line) and nonlinear model-based control with integral action implemented via simulation on the process model (dash-dotted line). The horizontal dotted line denotes the system pressure set-point ($P^{sp}_{sys} = 150$ psi).

(PI) controller with $\tau_r = 5$ leads to an extremely slow convergence to the set-point (on the order of 10 minutes). It is also seen that when a smaller integral time constant is used, it results in significant oscillations around the set-point due to the coupling between the two control loops. These oscillations cause large fluctuations in the feed flow rate (due to the VFD control loop) and could damage the feed pumps and cause fatigue on system components. Similar results are evident in Fig. 5. The application of the nonlinear controller to the experimental system causes the most deviation from the pressure set-point due to the speed at which it converges to the set-point. It can be seen that the PI controller causes almost no deviation from the set-point (approximately the same as the simulated nonlinear controller) because the convergence (change in valve position) is much slower. As the valve closes, it causes the system pressure to rise, forcing loop I to take action in order to keep the system pressure at the set-point. Slower valve actions allow more time for loop I to act and keep the system pressure at the set-point, such as in the case of the PI control with $\tau_r = 5$. Additional results from the experiments can be seen in the submitted journal paper (Bartman et al. (2009a)).

REFERENCES

Abbas, A. (2006). Model predictive control of a reverse osmosis desalination unit. *Desalination*, 194, 268–280.

Alatiqi, I., Ettourney, H., and El-Dessouky, H. (1999). Process control in water desalination industry: an overview. *Desalination*, 126, 15–32.

Alatiqi, I.M., Ghabris, A.H., and Ebrahim, S. (1989). System identification and control of reverse osmosis desalination. *Desalination*, 75, 119–140.

Assef, J.Z., Watters, J.C., Deshpande, P.B., and Alatiqi, I.M. (1997). Advanced control of a reverse osmosis desalination unit. *J. Proc. Cont.*, 4, 283–289.

Bartman, A.R., Christofides, P.D., and Cohen, Y. (2009a). Nonlinear model-based control of an experimental reverse osmosis water desalination system. *Industrial & Engineering Chemistry Research,* submitted.

Bartman, A.R., McFall, C.W., Christofides, P.D., and Cohen, Y. (2009b). Model predictive control of feed flow reversal in a reverse osmosis desalination process. *J. Process Contr.*, 19, 433–442.

Bird, R.B., Stewart, W.E., and Lightfoot, E.N. (2002). *Transport Phenomena, Second Edition.* John Wiley and Sons.

Burden, A.C., Deshpande, P.B., and Watters, J.C. (2001). Advanced control of a B-9 Permasep permeator desalination pilot plant. *Desalination*, 133, 271–283.

Chen, J., Wang, F., Meybeck, M., He, D., Xia, X., and Zhang, L. (2005). Spatial and temporal analysis of water chemistry records (19582000) in the Huanghe (Yellow River) basin. *Global Biogeochem. Cycles*, 19, GB3016.

Christofides, P.D. and El-Farra, N.H. (2005). *Control of Nonlinear and Hybrid Process Systems: Designs for Uncertainty, Constraints and Time-Delays,* 446 pages. Springer, New York.

El-Farra, N.H. and Christofides, P.D. (2001). Integrating robustness, optimality, and constraints in control of nonlinear processes. *Chemical Engineering Science*, 56, 1841–1868.

El-Farra, N.H. and Christofides, P.D. (2003). Bounded robust control of constrained multivariable nonlinear processes. *Chemical Engineering Science*, 58, 3025–3047.

Gambier, A. and Badreddin, E. (2002). Application of hybrid modeling and control techniques to desalination plants. *Desalination*, 152, 175–184.

Herold, D. and Neskakis, A. (2001). A small PV-driven reverse osmosis desalination plant on the island of gran canaria. *Desalination*, 137, 285–292.

Liu, C.C.K., Park, J., Migita, R., and Qin, G. (2002). Experiments of a prototype wind-driven reverse osmosis desalination system with feedback control. *Desalination*, 150, 277–287.

Lu, Y., Hu, Y., Zhang, X., Wu, L., and Liu, Q. (2007). Optimum design of reverse osmosis system under different feed concentration and product specification. *Journal of Membrane Science*, 287, 219–229.

McFall, C.W., Bartman, A.R., Christofides, P.D., and Cohen, Y. (2008). Control and monitoring of a high-recovery reverse-osmosis desalination process. *Industrial & Engineering Chemistry Research*, 47, 6698–6710.

Rahardianto, A., Gao, J., Gabelich, C.J., Williams, M.D., and Cohen, Y. (2007). High recovery membrane desalting of low-salinity brackish water: Integration of accelerated precipitation softening with membrane RO. *Journal of Membrane Science*, 289, 123–137.

Periodic Control of Gas-phase Polyethylene Reactors

Al-haj Ali, M., Ali, E.

Chemical Engineering Department, King Saud University
P.O.Box: 800, 11421 Riyadh, Saudi Arabia, (alhajali@ksu.edu.sa)

Abstract: Nonlinear model predictive control algorithm is used for the on-line control of polymer molecular weight distribution. The control of chain-length distribution is achieved by selecting a collection of points in the distribution and using it as set points for the control algorithm. An on-line Kalman filter is used to incorporate infrequent and delayed off-line molecular weight measurements. Through simulation; the control algorithm is evaluated, under tracking conditions as well as plant-model mismatch. The results demonstrate that the control algorithm can regulate the entire molecular weight distribution with high computational efficiency and minimum steady state error.

Keywords: Molecular weight distribution, nonlinear model predictive control, Kalman filter, polymerization reactor control, fluidized bed reactor, polyethylene

1. INTRODUCTION

Polymers today are crucial products that are used in all parts of our daily life. The range of applications includes standard applications as packaging materials and textile fibers, and special ones in the automobile and electrical industries. Molecular weight distribution (MWD) is considered as one of the fundamental properties that determines polymer properties and thus its applications. Therefore, it is important to monitor and control MWD during the industrial production of polymers. A significant amount of research has been done in the area of control, monitoring and modelling of polymerization reactors; Excellent reviews have been given by several researchers (Elicabe and Meira, 1988, Embirucu et al., 1996, Congalidis and Richards, 1998, Richards and Congalidis, 2006). A careful study of previous works, with focus on the description of polymer MWD, results in the following conclusions:

1. Most of the work that was done described polymer MWD by the weight average molecular weight (M_w), in addition to polydispersity index (PDI). Few researchers used the entire molecular weight distribution in their control studies. The use of Mw and PDI to describe polymer quality is helpful. However, sometimes, the molecular weight averages can be misleading when the molecular weight distribution shows bimodalities and/or it has high molecular weight tails. Besides, it is very useful to describe polymer quality by using the entire molecular weight distribution because in many polymer applications such as paints and paper coatings, it is required to specify such distribution properly (Sayer et al., 2001).

2. In polyolefins polymerization, the work done to control the entire MWD of the produced polymer used mixtures of different metallocenes (Chatzidoukas et al., 2007, Heiland and kaminsky, 1992) or a hybrid catalyst of Ziegler-Natta and metallocene catalysts in a one stage process (Shamshoum et al., 2003). The use of single reactor to produce the desired polymer is cost-efficient alternative. However, the mixture of different catalysts may lead to complex undesirable catalyst interactions and non-reproducible catalyst behaviour due to the high variability of the polymerization rate of each catalyst (Nele and Pinto, 2000). Additionally, the implementation of such catalyst systems requires a deep understanding of polymerization mechanisms using these catalysts; which is not a simple task. Finally, this method is still in the research phase and, it may take a long time before it can be (if it is developed successfully) widely implemented in industry. An alternative approach is to vary the polymerization conditions periodically in a single polymerization reactor. The periodic operation of continuous chemical reactors can improve the performance of the reacting system and allow better design and control of the molecular weight distribution in a single reactor (Nele and Pinto, 2000, Schiffino, 1995).

The scope of this work is to investigate the production of polyethylene, in a fluidized bed reactor, with a well-defined molecular weight distribution using nonlinear model predictive controller (NLMPC).

2. PROCESS MODEL

In fluidized-bed polyethylene reactors, the co-polymerization of ethylene and -olefin monomers is carried out using a multi-site Ziegler-Natta catalyst, which consists of three different types of active sites. Each active site produces polymer with molecular weight distribution that can be described by Schulz-Flory distribution. The polyethylene reactor process is depicted in Fig 1. The process model was

developed by (McAuley et al., 1995), modifications made in this model were described in (Ali et al., 2003).

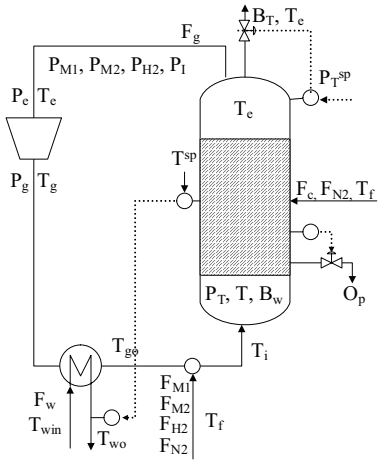


Fig. 1. Polyethylene reactor.

2.1 Molecular Weight Distribution Model

The instantaneous molecular weight distribution for each type of active sites can be described by Flory-Schulz exponential function (Kissin et al., 2005)

$$y_j^d = j \cdot q^2 \cdot \exp(-j \cdot q) \tag{1}$$

with q is the termination probability, j is the number of repeating units and y_j^d instantaneous weight distribution. As assumed above, the catalyst consists of three different active sites and the distribution of the polymers produced by each site type can be represented by Flory's most probable distribution. Thus, the overall distribution of the produced polymer can be calculated by the weighted sum of the three distributions as given below

$$y_{j,ins} = \sum_{i=1}^{3} w_i \cdot (y_j^d)_i \tag{2}$$

where $y_{j,ins}$ is the overall instantaneous molecular weight distribution, and w_i is the mass fraction of each site. The molecular weight distribution of the polymer accumulated in the reactor after a certain polymerization time can be calculated using the following equation:

$$\frac{dy_j}{dt} = \frac{O_p \cdot (y_{j,ins} - y_j)}{B_w} \tag{3}$$

here y_j is the cumulative molecular weight distribution, O_p is polymer production rate and B_w is mass of polymer in the reactor bed. Finally the GPC reading of the MWD is calculated by the following equation:

$$GPC = j \cdot y_j \cdot ln(10) \tag{4}$$

3. ON-LINE NLMPC ALGORITHM

In this work, the structure of the MPC version developed by Ali and Zafiriou (1993) that utilizes directly the nonlinear model for output prediction is used. A usual MPC formulation solves the following on-line optimization:

$$\min_{\Delta u(t_k),\ldots,\Delta u(t_{k+M-1})} \sum_{i=1}^{P} \left\| \Gamma(y(t_{k+i}) - R(t_{k+i})) \right\|^2 + \sum_{i=1}^{M} \left\| \Lambda \, \Delta u(t_{k+i-1}) \right\|^2 \tag{5}$$

subject to

$$A^T \Delta U(t_k) \le b \tag{6}$$

For nonlinear MPC, the predicted output, y over the prediction horizon P is obtained by the numerical integration of:

$$\frac{dx}{dt} = f(x,u,t) \tag{6}$$

$$y = g(x) \tag{7}$$

from t_k up to t_{k+P} where x and y represent the states and the output of the model, respectively. The symbols || . || denotes the Euclidean norm, k is the sampling instant, Γ and Λ are diagonal weight matrices and R = [r(k+1) … r(k+P)]^T is a vector of the desired output trajectory. $\Delta U(t_k)=[\Delta u(t_k)$ … $\Delta u(t_{k+M-1})]^T$ is a vector of M future changes of the manipulated variable vector u that are to be determined by the on-line optimization. The control horizon (M) and the prediction horizon (P) are used to adjust the speed of the response and hence to stabilize the feedback behavior. Γ is usually used for trade-off between different controlled outputs. The input move suppression, Λ, on the other hand, is used to penalize different inputs and thus to stabilize the feedback response. The objective function (Eq. 5) is solved on-line to determine the optimum value of $\Delta U(t_k)$. Only the current value of Δu, which is the first element of $\Delta U(t_k)$, is implemented on the plant. At the next sampling instant, the whole procedure is repeated.

To compensate for modeling error and eliminate steady state offset, a regular feedback is incorporated on the output predictions, $y(t_{k+1})$ through an additive disturbance term. Therefore, the output prediction is corrected by adding to it the disturbance estimates. The latter is set equal to the difference between plant and model outputs at present time k as follows:

$$d(k) = y_p(k) - y(k) \tag{7}$$

The disturbance estimate, d, is assumed constant over the prediction horizon due to the lack of an explicit means of predicting the disturbance. However, for severe modeling errors, or open-loop unstable processes the regular feedback is not enough to improve the NLMCP response. Hence, state or parameter estimation is necessary to enhance the NLMPC performance in the face of model-plant mismatch. In this work, Kalman filtering (KF) will be incorporated to correct the model state and thus, to address the robustness issue. Utilization of the NLMPC with KF requires adjusting an additional parameter, σ. More details on the integration of KF

with the NLMCP algorithm are given elsewhere (Ali and Zafiriou, 1993). In addition to state estimation by KF, the predicted output will be also corrected by the additive disturbance estimates of Eqn.7.

The main objective of the NLMPC is to control the entire MWD. It is also necessary to maintain acceptable polymer production rate. Process stability is another important issue which is handled through regulating the total gas pressure and the bed temperature. These two controlled variables are adapted via separate PI control loops. The design and tuning parameters of these loops are given elsewhere (Ali et al., 2003).

4. RESULTS AND DISCUSSION

It is worth mentioning that determining input trajectories that provide the desired distribution is difficult as the final polymer quality is sensitive to hydrogen concentration (X) value and the mass of the produced polymer. In this sense, maintaining the desired MWD during process operation is even more challenging. In the presence of model-plant mismatch and/or when unmeasured disturbances enter the plant, the situation becomes more complex. The control objective here is to produce broad polyethylene with well-defined MWD starting from narrow distribution and maintain it there. The results of this case are shown in Figs. 2 and 3. Four manipulated variables, which are the monomer, hydrogen, nitrogen and catalyst flow rates, are used. The weighting factors for these inputs are Λ=[0 0 20 50]. Four controlled variables, which represent specific points in the target MWD, are considered as shown by the dots in Fig 3. The weighting factor for all outputs is given the same value of Γ=[1 1 1 1]×100. The lower limit for the manipulated variables is set to zero and the upper limit is set to twice their nominal values. The MWD target function contains 103 points, however only four points were selected as controlled outputs to reduce the computation effort consumed by the NLMPC calculations. The input horizon (M) and output horizon (P) are taken equal to 1 and 4, respectively. A sampling time of 1 hr is used. Usually the GPC measurements are available at low frequency. Advanced measurement sensors that can provide measurements in the order of minutes are available but at high cost.

Fig. 3 demonstrates the ability of NLMPC to maintain the new set point for the polymer distribution with minor distortion in the distribution function. More interesting is the response of the manipulated variables as shown in Fig. 2. The resulted response of the manipulated variables is in the form of periodic functions. Long prediction and moving horizon capability of NLMPC helped the controller to understand the dynamic nature of the process to an extent that it produced cyclic input sequences. Moreover, Fig. 2 shows how the bleed flow rate (B_T) and the cooling water inlet temperature (T_w) varies by separate PI controllers to maintain the total pressure at 20 atm and the reactor temperature at 82 ^oC. Note that the manipulated variables used by NLMPC are plotted in discrete form because the NLMPC works in discrete time fashion.

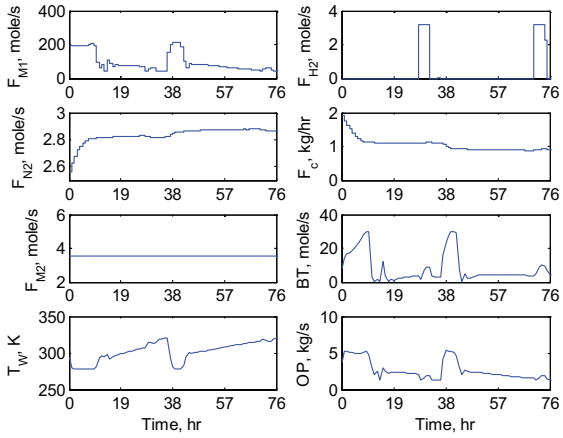


Fig. 2. Manipulated variable response using NLMPC.

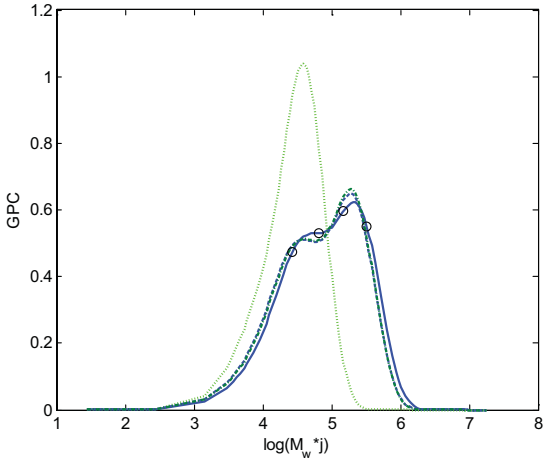


Fig. 3. MWD using NLMPC. Dotted line: initial distribution, solid: target, dashed: controlled distribution.

Next the algorithm was tested for targeting another MWD. In this case, seven points on the GPC curve is taken as the controlled variables with their weights are fixed at G=[1 100 100 200 100 50] ×102. The lower limit of F_{M1} is set to 40 mole/s to keep high monomer concentration in the reactor. The value of the rest of the parameters remains the same as before. The simulation results are shown in Figs. 4 - 5. Evidently, NLMPC generated suitable periodic input sequences that produce MWD close to the desired one as shown in Fig. 5. The MWD suffered from minor distortion; however exact match of the target function is not necessary especially when we know that the relative error in GPC measurements is around 10%. This outcome can be obtained at shorter simulation time. The small production rate is obvious from Fig. 4; in fact, the average production rate is found to be 2.42 kg/s. To improve the production rate, the latter is incorporated as a controlled variable in the NLMPC algorithm. Using γ=0.1 for the production rate, NLMPC managed to increase the polymer production to 2.86 kg/s but with notable loss of the MWD. Results are not shown here for simplicity. Increasing the weight of the designated

controlled output further will of course propagates the production rate but the MWD will depart away from the desired set point. Our investigation revealed the existence of trade-off between the production rate and broadening the MWD. Widening the distribution requires pronounced changes in hydrogen concentration inside the reactor. Increasing hydrogen concentration is achieved by feeding more hydrogen to the reactor this reduces ethylene polymerization rate and as a consequence reduces the overall production rate. Whereas, reducing hydrogen concentration is achieved by opening the vent (Lo and Ray, 2006) that allows hydrogen concentration in the reactor to escape, causing hydrogen concentration to fall quickly. Such reduction in the concentration affects positively on the production rate.

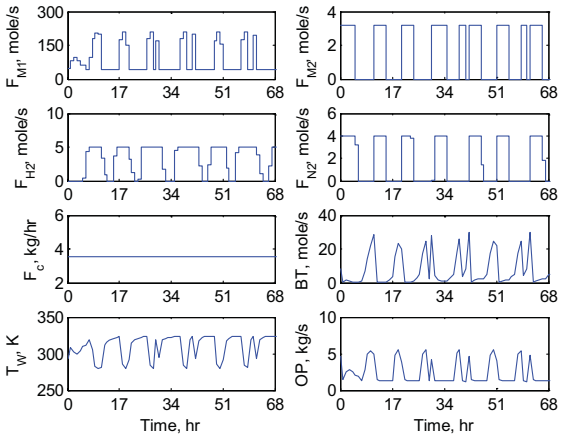


Fig. 4. Manipulated variable response using NLMPC. Decreasing polymer average molecular weight.

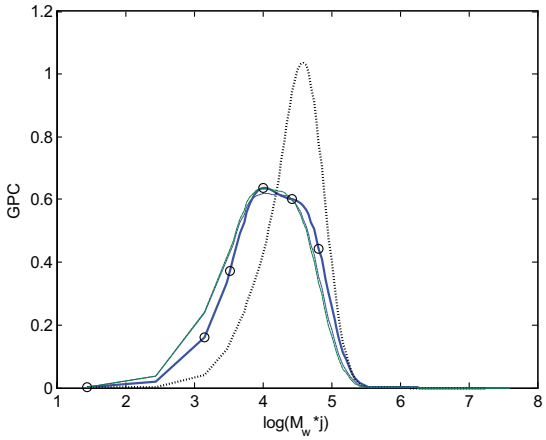


Fig. 5. MWD using NLMPC. Dotted line: initial distribution, solid: target, dashed: controlled distribution.

The previous simulations are carried out assuming perfect model. However, this is not always true in real practice. To test the robustness of NLMPC to reject the effect of modeling errors, the simulation of targeting higher molecular weight is repeated with−20% error in the reaction rate constant and catalyst activity. The results are shown in Figs. 6 - 7. It is evident that NLMPC is able to keep good control performance despite minor loss of controller performance.

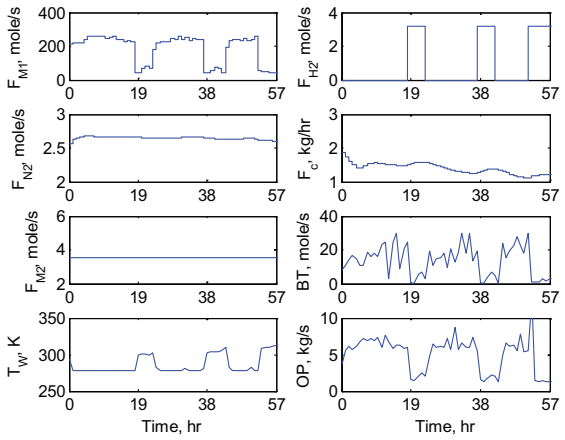


Fig. 6. Manipulated variable response using NLMPC in the presence of -20% in catalyst activation and reaction rate constant.

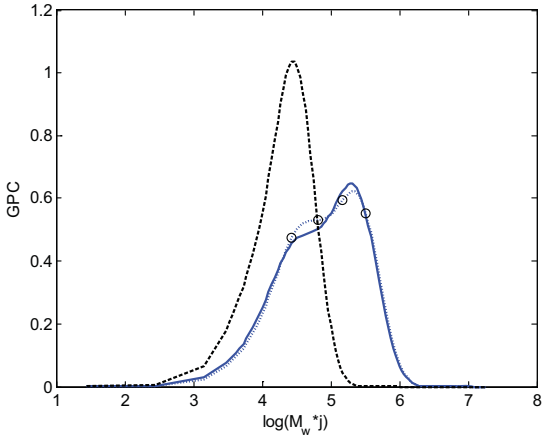


Fig. 7. MWD using NLMPC in the presence of -20% error in catalyst activation and reaction rate constant. Dotted line: initial distribution, solid: target, dashed: controlled distribution.

It is worth mentioning that controller performance could be improved more if the dynamics of hydrogen is faster. Since, hydrogen is not consumed in the reactor and large fluctuations in hydrogen concentration are required to broaden polymer distribution, improving controller performance would not be an easy task. This challenge can be solved using either a catalyst that is highly-sensitive to hydrogen as metallocenes or hydrogen consuming agent. The first approach depends on implementing a relatively new catalyst that is not widely used industrially (Galli and Vecellio, 2001). The second approach still needs more investigation to prove its applicability for the studied process. Finally, note that venting is usually used to reduce hydrogen concentration, as described above, however; venting reactor contents is not an economical choice because monomer also

escapes from the reactor. Nonetheless, no other choices are available.

5. CONCLUSIONS

In industrial applications, the molecular weight distribution of the produced polymer is usually measured using molecular weight averages and polydispersity index. In this article, we have presented an on-line MWD control technique to produce polymers with a target distribution in a fluidized-bed polymerization process. This strategy uses detailed polymerization process model, and Kalman filter to correct model states. A NLMPC controller is designed to control polymer MWD and polymerization process productivity. For the calculation of the MWD, selected points in polymer distribution curve are used as set-points for the controller that manipulates monomer, hydrogen, nitrogen and catalyst feed rates. To test the feasibility of the proposed MWD control technique, simulations have been carried out for ethylene gas-phase polymerization using conventional Zielger-Natta catalyst. The simulations suggest that the proposed control strategy can be useful new technique to control the MWD of polymer in continuous polymerization processes. The performance of the developed control algorithm can be improved more if the dynamic response of hydrogen concentration inside the polymerization reactor is less sluggish.

ACKNOWLEDGMENT

The financial support from Saudi Basic Industries Company (SABIC), (grant number 7/429), is greatly appreciated.

REFERENCES

Ali, E. M., Al-Humaizi, K. and Ajbar, A. (2003).Multivariable Control of a Simulated Industrial Gas-Phase Polyethylene Reactor. *Industrial and Engineering Chemistry Research,* **42,** 2349-2364.

Ali, E. M. and Zafiriou, E. (1993).Optimization-based Tuning of Non-linear Model Predictive Control with State Estimation. *J. of Process Control,* **3,** 97-107.

Chatzidoukas, C., Kanellopoulos, V. and Kiparssides, C. (2007).On The Production of Polyolefins with Bimodal Molecular Weight and Copolymer Composition Distributions in Catalytic Gas-phase Fluidized-bed Reactors. *Macromolecular Theory and Simulation,* **16,** 755-769.

Congalidis, J. P. and Richards, J. R. (1998).Process control of polymerization reactors: An industrial perspective. *Polymer Reaction Engineering,* **6,** 71-111.

Elicabe, G. E. and Meira, G. R. (1988).Estimation and Control in Polymerization Reactors. A Review. *Polymer Engineering and Science,* **28,** 121-135.

Embirucu, M., Lima, E. L. and Pinto, J. C. (1996).A Survey of Advanced Control of Polymerization Reactors. *Polymer Engineering and Science,* **36,** 433-447.

Galli, P. and Vecellio, G. (2001).Technology: Deriving Force Behind Innovation and Growth of Polyolefins. *Progress in polymer Science,* **26,** 1287-1336.

Heiland, K. and kaminsky, W. (1992).Comparison of Zirconocene and Hafnocene Catalysts for the Polymerization of Ethylene and 1-Butene. *Makromoleculare Chemie-Macromolecular Chemistry and Physics,* **193,** 601-610.

Kissin, Y. V., Mirabella, F. M. and Meverden, C. C. (2005).Multi-Center Nature of Heterogeneous Ziegler–Natta Catalysts: TREF Confirmation. *Journal of Applied Polymer Science,* **43,** 4351-4362.

Lo, D. P. and Ray, W. H. (2006).Dynamic Modeling of Polyethylene Grade Transitions in Fluidized bed Reactors Employing Nickel-Diimine Catalysts. *Industrial and Engineering Chemistry Research,* **45,** 993-1008.

McAuley, K. B., McDonald, D. A. and McLellan, P. J. (1995).Effects of Operating Conditions on Stability of Gas-phase Polyethylene Reactors. *AIChE,* **41,** 868-879.

Nele, M. and Pinto, J. C. (2000).Retrofitting of Industrial Olefin Polymerization Plants: Producing Broad MWDs Through Multiobjective Periodic Operation. *Journal of Applied Polymer Science,* **77,** 437-452.

Richards, J. R. and Congalidis, J. P. (2006).Measurement and Control of Polymerization Reactors. *Computers and Chemical Engineering,* **30,** 1447-1463.

Sayer, C., Arzamendi, G., Asua, J. M., Lima, E. L. and Pinto, J. C. (2001).Dynamic Optimization of Semicontinuous Emulsion Copolymerization Reactions: Composition and Molecular Weight Distribution. *Computers and Chemical Engineering,* **25,** 839-849.

Schiffino, R. S. (1995) USA.

Shamshoum, E. S., Chen, H. and Margarito, L. (2003) USA.

Control of Nonlinear System – Adaptive and Predictive Control

Jiri Vojtesek*, Petr Dostal*, Vladimir Bobal*

**Department of Process Control, Faculty of Applied Informatics,*
Tomas Bata University in Zlin, Czech Republic
(Tel: 00420576035199; e-mail: {vojtesek,dostalp,bobal}@fai.utb.cz)

Abstract: The goal of this paper is to propose suitable control methods for controlling of the highly nonlinear system represented by the mathematical model of the continuous stirred tank reactor (CSTR) with so called van der Vusse reaction inside. Temperature of the reactant is controlled by the heat removal of the cooling liquid in the reactor's jacket. Two control strategies were suggested – adaptive control and predictive control. The adaptive approach uses recursive identification for the optimal setting of the controller. The predictive control computes input sequence by the minimizing of the cost function constructed by the difference between output variable and reference signal. Both control strategies shows good control results and pertinence for the controlling of such type of systems.

Keywords: Adaptive control, Predictive Control, Polynomial methods, Recursive estimation, Nonlinear systems, CSTR, Simulation.

1. INTRODUCTION

Unfortunately, most of the processes in the technical praxis have nonlinear properties. Typical example of the nonlinear system can be found in the chemical or the biochemical industry where so called chemical reactor is used for production of the several chemicals or drugs (Corriou, 2004).

Controlling of these devices with the conventional methods where parameters of the controller are set at the beginning fixed during the control could result in non-optimal control responses because of changing parameters of the system. This inconvenience could be overcome with use of other control strategies which takes into account these changes, for example adaptive or predictive control. These two control strategies are compared in this paper in order to compare obtained simulation results.

The basic idea of adaptive control is that parameters or the structure of the controller are adapted to parameters of the controlled plant according to the selected criterion (Bobál *et al.*, 2005). Adaptation can be done for example by the modification of the controller's parameters by the change of the controller's structure or by generating an appropriate input signal, which is called "adaptation by the input signal".

The polynomial synthesis (Kučera, 1993) is one of the methods used in adaptive control for control synthesis of the system. This method is based on the input-output model of the controlled system or its transfer function. It can be classified as an algebraic method and is based on algebraic operations in the ring of polynomials. Polynomials are usually described in s-plane for continuous systems, in z-plane for discrete systems and in δ-plane for systems which come from δ-models of both the controlled system and the controller too (Middleton and Goodwin, 2004) and (Mukhopadhyay *et al.*, 1992).

One of the biggest advantages of the polynomial method compared to the conventional method is that it provides not only relations for computing of the controller's parameters but the structure of the controller too. This structure fulfils general requirements for control systems and input signals (reference signal and disturbance) and it can be used for controlling of the systems with negative properties from the control point of view, such as non-minimum phase systems or unstable systems. Another advantage is that the resulted relations are easily programmable.

Polynomials in the numerator and denominator of the transfer function of the controller result from the solution of Diophantine equations, which have so called characteristic polynomial of the closed loop system on the right side of the equation. The roots of this polynomial are then poles of the closed-loop system, which affects the quality of control. The method of choosing the poles is called Pole-placement or Pole-assignment (Kučera, 1991).

The idea of the predictive control is based on the calculation of the control sequence from the actual time point minimizing the deviation of the reference signal and the output signal of the plant in the future horizon (Clarke *et al.*, 1987). The future values of the reference signal are given in advance or are assumed to be equal to the present one. The future values of the plant can be predicted from a process model. If disturbances are measurable, then their future values are predicted using some assumptions.

All approaches are verified by the simulation in the simulation program Matlab®, version 6.5.

2. ADAPTIVE CONTROL

The adaptive approach in this work is based on choosing an external linear model (ELM) of the original nonlinear system

whose parameters are recursively identified during the control. Parameters of the resulted continuous controller are recomputed in every step from the estimated parameters of the ELM (Bobál *et al.*, 2005).

2.1 External Linear Model

The main types of ELM are continuous-time (CT) models and discrete-time (DT) models.

The general description of the CT ELM can be formulated via transfer function $G(s)$:

$$G(s)=\frac{b(s)}{a(s)}=\frac{Y(s)}{U(s)}=\frac{b_m s^m+b_{m-1}s^{m-1}+\ldots+b_1 s+b_0}{a_n s^n+a_{n-1}s^{n-1}+\ldots+a_1 s+a_0} \tag{1}$$

The continuous-time ELM is supposed to be more accurate and corresponding to the real model because data are estimated continuously during the control. On the contrary, CT identification is difficult.

On the other hand, identification of the DT models are easy to realize. We can say that discrete models are used in the cases where the usage of continuous ones is complicated or the realization is impossible. An important variable in the discrete-time models is sampling period T_v.

The transfer function G in this case is defined as Z-transform of the output variable y to the input variable u

$$G(z)=\frac{Y(z)}{U(z)}=\frac{b(z)}{a(z)}=\frac{b_m z^m+b_{m-1}z^{m-1}+\ldots+b_1 z+b_0}{a_n z^n+a_{n-1}z^{n-1}+\ldots+a_1 z+a_0} \tag{2}$$

where $a(z)$ and $b(z)$ are discrete polynomials and $U(z)$ and $Y(z)$ are Z-transform images of the input and output variables.

2.2 Identification

The use of the discrete model for nonlinear system can cause problems with the sampling period T_v. This sampling period cannot be small because of the stability and the big sampling period is unacceptable because we do not know what will happen with the system during this sample.

The inconvenience with the sampling period could be overcome with the use of so called delta (δ-) models. Although the delta operator belongs to the class of discrete models with the operator described as

$$q\cdot x(k)\triangleq x(k+1) \tag{3}$$

it can been seen from

$$\delta=\frac{q-1}{T_v} \tag{4}$$

that this operator is related to sampling period T_v and it means that δ-models are close to the continuous ones in d/dt.

A new complex variable in "δ" plane called "γ", which is defined for example in (Mukhopadhyay *et al.*, 1992) as

$$\gamma=\frac{z-1}{\beta\cdot T_v\cdot z+(1-\beta)\cdot T_v} \tag{5}$$

need to be introduced. We can obtain an infinite number of δ–models for different values of optional parameter β in Equation (5) from the range $0\le\beta\le 1$.

The forward δ-model described for $\beta = 0$ by

$$\gamma=\frac{z-1}{T_v} \tag{6}$$

is dealt with in this work.

The Recursive Least-Squares (RLS) method is used for the parameter estimation in this work. The RLS method is well-known and widely used for the parameter estimation (Fikar and Mikleš, 1999). It is usually modified with some kind of forgetting, exponential or directional (Kulhavý and Karny, 1984), because parameters of the identified system can vary during the control which is typical for nonlinear systems and the use of some forgetting factor could result in better output response.

The RLS method with exponential forgetting is describe by the set of equations:

$$\begin{aligned}
&\varepsilon(k)=y(k)-\varphi^T(k)\cdot\hat{\theta}(k-1)\\
&\gamma(k)=\left[1+\varphi^T(k)\cdot\mathbf{P}(k-1)\cdot\varphi(k)\right]^{-1}\\
&\mathbb{L}(k)=\gamma(k)\cdot\mathbf{P}(k-1)\cdot\varphi(k)\\
&\mathbf{P}(k)=\frac{1}{\lambda_1(k-1)}\left[\mathbf{P}(k-1)-\frac{\mathbf{P}(k-1)\cdot\varphi(k)\cdot\varphi^T(k)\cdot\mathbf{P}(k-1)}{\lambda_1(k-1)+\varphi^T(k)\cdot\mathbf{P}(k-1)\cdot\varphi(k)}\right]\\
&\hat{\theta}(k)=\hat{\theta}(k-1)+\mathbb{L}(k)\varepsilon(k)
\end{aligned} \tag{7}$$

Several types of exponential forgetting can be used, e.g. like RLS with constant exponential forgetting, RLS with increasing exp. forgetting etc. RLS with the changing exp. forgetting is used for parameter estimation, where the changing forgetting factor λ_1 is computed from the equation

$$\lambda_1(k)=1-K\cdot\gamma(k)\cdot\varepsilon^2(k) \tag{8}$$

Where K is small number, in our case $K = 0.001$.

2.3 Polynomial Synthesis

The structure of the controller is designed via polynomial synthesis. The simple one degree-of-freedom (1DOF) control configuration was used. The block scheme of this configuration in Fig. 1.

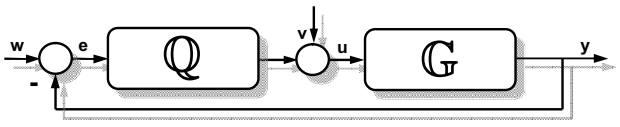


Fig. 1 1DOF control configuration

Block Q in Fig. 1 represents the transfer function of the controller, G denotes the transfer function of the plant, w is

the reference signal, e is used for the control error, v is the disturbance at the input to the system, u determines the input variable, and finally y is the output variable.

Transfer functions of the controller and controlled plant could be described in the continuous time by equations:

$$Q(s)=\frac{q(s)}{s\cdot p(s)};\ G(s)=\frac{b(s)}{a(s)} \tag{9}$$

where polynomials $p(s)$ and $q(s)$ are designed by the polynomial approach and parameters of these polynomials are computed by the Method of uncertain coefficients which compares coefficients of individual s-powers from Diophantine equation (Kučera, 1993):

$$a(s)\cdot s\cdot p(s)+b(s)\cdot q(s)=d(s) \tag{10}$$

Although parameters of the polynomials $a(s)$ and $b(s)$ are reflected to be in continuous-time, the identification runs recursively in discrete time periods related to the sampling period T_v. This simplification is supported by the use of δ-models where each input and output variable is recomputed to this sampling period, T_v, which shifts these discrete polynomials closer to the continuous ones. It was proofed for example in (Stericker and Sinha, 1993) that the parameters of the delta model for the small sampling period approach to the continuous ones in (9).

The feedback controller $Q(s)$ in Fig. 1 ensures all basic control requirements – i.e. stability, load disturbance attenuation and asymptotic tracking of the reference signal. It is required that each controller could be tuned somehow. This option can be found in this controller in the stable optional polynomial $d(s)$ on the right side of the Diophantine equation (10). As it is mentioned above, there are several methods for choosing of this polynomial. The method used here is Pole-placement or Pole-assignment method. Polynomial $d(s)$ can be divided into two parts – $m(s)$ and $n(s)$, so

$$d(s)=m(s)\cdot n(s) \tag{11}$$

where polynomial $n(s)$ is computed from the spectral factorization of polynomial $a(s)$ in the denominator of the transfer function $G(s)$ (9)

$$n^*(s)\cdot n(s)=a^*(s)\cdot a(s) \tag{12}$$

and polynomial $m(s)$ is a stable one $m(s)=(s+\alpha_i)^{\deg d-\deg n}$ and $\alpha_i>0$ are (deg d – deg n) optional stable roots, usually called poles of the control system. A disadvantage of this method can be found in the uncertainty of the polynomial $m(s)$ – there is no general rule how to choose roots α_i.

3. PREDICTIVE CONTROL

3.1 Generalized Predictive Control

Generalized Predictive Control (GPC) is one of the most popular predictive methods based on Model Predictive Control (MPC) (Clarke *et al.*, 1987), and has been successfully used in praxis for different types of control problems from this time.

The GPC has many common ideas with the ordinary predictive methods but it has some differences to such as the solution of the GPC controller is analytical, it can be used for unstable and non-minimum phase systems etc.

The general single-input single-output (SISO) after linearization can be described through the discrete backshift operators z^{-1} as

$$A(z^{-1})\cdot y(t)=z^{-d}\cdot B(z^{-1})\cdot u(t-1)+C(z^{-1})\cdot e(t) \tag{13}$$

where $u(t)$ is control variable, $y(t)$ output variable, $e(t)$ denotes a zero mean white noise, and d is dead time of the system. Polynomials $A(z^{-1})$, $B(z^{-1})$ and $C(z^{-1})$ are

$$\begin{aligned}A(z^{-1})&=1+a_1z^{-1}+a_2z^{-2}+\ldots+a_{n_a}z^{-n_a}\\ B(z^{-1})&=b_0+b_1z^{-1}+b_2z^{-2}+\ldots+b_{n_b}z^{-n_b}\\ C(z^{-1})&=1+c_1z^{-1}+c_2z^{-2}+\ldots+c_{n_c}z^{-n_c}\end{aligned} \tag{14}$$

Equation (23) is called the *Controller Auto-Regressive Moving-Average* (CARMA) *model*. This model is not suitable in most industrial processes where disturbances are non-stationary. In these cases, the integrated CARMA (CARIMA) model is more suitable

$$A(z^{-1})\cdot y(t)=z^{-d}\cdot B(z^{-1})\cdot u(t-1)+C(z^{-1})\cdot\frac{e(t)}{\Delta} \tag{15}$$

where $\Delta=1-z^{-1}$.

The GPC algorithm can be then formulated as minimization of the cost function

$$\begin{aligned}J_{GPC}=&\sum_{j=N_1}^{N_2}\delta_u(j)\left[\hat{y}(t+j|t)-w(t+j)\right]^2+\ldots\\ &\ldots+\sum_{j=1}^{N_u}\lambda_u(j)\left[\Delta u(t+j-1)\right]^2\end{aligned} \tag{16}$$

where $\hat{y}(t+j|t)$ is an optimum j-ahead prediction of the output on data up to time t, further, N_1 and N_2 denote minimum and maximum costing horizons, respectively, N_u is control horizon, $w(t+j)$ means reference signal, Δu stands for manipulated variable and finally $\delta_u(j)$ and $\lambda_u(j)$ denote weighting sequences.

The values of these factors are for simplification assigned as $\delta_u=1$, and λ_u is constant through the whole time interval of the control.

4. MODEL OF THE PLANT

The nonlinear system under the consideration is the Continuous Stirred Tank Reactor (CSTR). The reaction inside the reactor is called *van der Vusse* reaction can be described by the following reaction scheme (Chen, *et al.*, 1995):

$$A \xrightarrow{k_1} B \xrightarrow{k_2} C$$
$$2A \xrightarrow{k_3} D \quad (17)$$

The graphical scheme of this reactor can be seen in Fig. 2

The mathematical model of this reactor is described by the following set of ordinary differential equations (ODE):

$$\frac{dc_A}{dt} = \frac{q_r}{V_r}\left(c_{A0} - c_A\right) - k_1 c_A - k_3 c_A^2 \quad (18)$$

$$\frac{dc_B}{dt} = -\frac{q_r}{V_r} c_B + k_1 c_A - k_2 c_B \quad (19)$$

$$\frac{dT_r}{dt} = \frac{q_r}{V_r}\left(T_{r0} - T_r\right) - \frac{h_r}{\rho_r c_{pr}} + \frac{A_r U}{V_r \rho_r c_{pr}}\left(T_c - T_r\right) \quad (20)$$

$$\frac{dT_c}{dt} = \frac{1}{m_c c_{pc}}\left(Q_c + A_r U\left(T_r - T_c\right)\right) \quad (21)$$

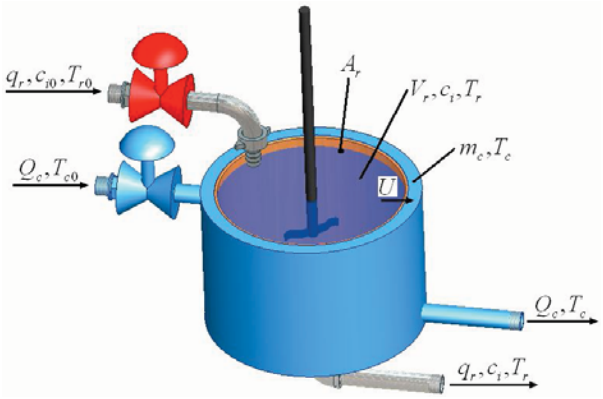


Fig. 2: Continuous Stirred Tank Reactor (CSTR)

This set of ODE together with simplifications then mathematically represents examined CSTR reactor. The model of the reactor belongs to the class of *lumped-parameter nonlinear systems*. Fixed parameters of the system are shown in Table 1 (Chen, *et al*., 1995).

Table 1 Fixed parameters of CSTR

$k_{01} = 2.145 \cdot 10^{10}\ min^{-1}$	$k_{02} = 2.145 \cdot 10^{10}\ min^{-1}$
$k_{03} = 1.5072 \cdot 10^{8} min^{-1}.mol^{-1}$	$E_1/R = 9758.3\ K$
$E_2/R = 9758.3\ K$	$E_3/R = 8560\ K$
$h_1 = -4200\ kJ.kmol^{-1}$	$h_2 = 11000\ kJ.kmol^{-1}$
$h_3 = 41850\ kJ.kmol^{-1}$	
$V_r = 0.01\ m^3$	$\rho_r = 934.2\ kg.m^{-3}$
$c_{pr} = 3.01\ kJ.kg^{-1}.K^{-1}$	$q_r = 2.365 \cdot 10^{-3}\ m^3 min^{-1}$
$c_{pc} = 2.0\ kJ.kg^{-1}.K^{-1}$	$Q_c = -18.5583\ kJ.min^{-1}$
$U = 67.2\ kJ.min^{-1}m^{-2}K^{-1}$	$A_r = 0.215\ m^2$
$c_{A0} = 5.1\ kmol.m^{-3}$	$c_{B0} = 0\ kmol.m^{-3}$
$T_{r0} = 387.05\ K$	$m_c = 5\ kg$

The reaction heat (h_r) in eq. (20) is expressed as:

$$h_r = h_1 \cdot k_1 \cdot c_A + h_2 \cdot k_2 \cdot c_B + h_3 \cdot k_3 \cdot c_A^2 \quad (22)$$

where h_i means reaction enthalpies.

Nonlinearity can be found in reaction rates (k_j) which are described via Arrhenius law:

$$k_j\left(T_r\right) = k_{0j} \cdot \exp\left(\frac{-E_j}{RT_r}\right), \text{ for } j = 1,2,3 \quad (23)$$

where k_0 represent pre-exponential factors and E are activation energies.

Static analysis has shown (Vojtesek, *et al.*, 2004), that system has an optimal working point for volumetric flow rate of the reactant $q_r = 2.365 \times 10^{-3}\ m^3.min^{-1}$ a heat removal $Q_c = -18.56\ kJ.min^{-1}$. The difference between actual and initial temperature of the reactant T_r was taken as controlled output and changes of the heat removal Q_c was set as control input, i.e.

$$y(t) = T_r(t) - T_r^s(t)\,[K]$$
$$u(t) = 100 \cdot \frac{Q_c(t) - Q_c^s(t)}{Q_c^s(t)}\,[\%] \quad (24)$$

On the other hand, dynamic analysis results in ELM represented by a second order transfer function with relative order one, which is generally:

$$G(s) = \frac{b(s)}{a(s)} = \frac{b_1 s + b_0}{s^2 + a_1 s + a_0} \quad (25)$$

Equation (25) can be rewritten for the identification to the form of the differential equation

$$y_\delta(k) = -a_1 y_\delta(k-1) - a_0 y_\delta(k-2) + b_1 u_\delta(k-1) + b_0 u_\delta(k-2) \quad (26)$$

where y_δ is recomputed output to the δ-model:

$$y_\delta(k) = \frac{y(k) - 2y(k-1) + y(k-2)}{T_v^2}$$
$$y_\delta(k-1) = \frac{y(k-1) - y(k-2)}{T_v} \quad y_\delta(k-2) = y(k-2) \quad (27)$$
$$u_\delta(k-1) = \frac{u(k-1) - u(k-2)}{T_v} \quad u_\delta(k-2) = u(k-2)$$

where T_v is the sampling period, the data vector is

$$\phi^T(k-1) = \left[-y_\delta(k-1), -y_\delta(k-2), u_\delta(k-1), u_\delta(k-2)\right] \quad (28)$$

and the vector of estimated parameters

$$\hat{\Theta}^T(k) = \left[\hat{a}_1, \hat{a}_0, \hat{b}_1, \hat{b}_0\right] \quad (29)$$

could be computed from the ARX (Auto-Regressive eXogenous) model

$$y_\delta(k) = \hat{\Theta}^T(k)\phi(k-1) \quad (30)$$

by the recursive least squares methods described in part 2.2.

The ELM is of the second order, which means that degrees of polynomials $p(s)$, $q(s)$, and $d(s)$ are then:

$$\deg q = 2; \deg p = 1; \deg d = 4 \quad (31)$$

and polynomials $m(s)$ and $n(s)$ in the equation (11) are

$$n(s) = s^2 + n_1 s + n_0; \quad m(s) = \left(s + \alpha_i\right)^{\deg d - \deg n} = \left(s + \alpha_i\right)^2 \quad (32)$$

and coefficients of the polynomial $n(s)$ are computed via spectral factorization (12) as

$$n_0 = \sqrt{a_0^2}, n_1 = \sqrt{2n_0 + a_1^2 - 2a_0} \tag{33}$$

Transfer functions of the feedback and feedforward parts of the controller for 1DOF and 2DOF configurations are

$$Q(s) = \frac{q_2 s^2 + q_1 s + q_0}{s(s + p_0)} \tag{34}$$

Where parameters of the polynomials $q(s)$ and $p(s)$ by the comparison of the coefficients of the s-powers a in diophantine equations (10).

The identified parameters of the delta ELM from the predictive control were recomputed to the discrete-time ELM:

$$G(z^{-1}) = \frac{-0.0021 z^{-1} + 0.0010 z^{-2}}{1 - 1.5851 z^{-1} + 0.6197 z^{-2}} \tag{35}$$

which was used for computation of the predictive controller in Equation (25). Zero mean white noise $e(t)$ is not take into account.

5. SIMULATION RESULTS

Both control strategies were verified by simulation experiments. The sampling period was set to $T_v = 0.3$ *min*, all simulations took 500 min and 5 different step changes $w = [2, -1, 1\ -1, 1.5]$ were done during this interval.

The first simulation study was done for adaptive controller with the various values of the root $\alpha_i = 0.05$, 0.1 and 0.2 in Equation (32). The predictive controller was verified again for different values of the weighting factor $\lambda_u = 0.05$, 0.5 and 2 in (16).

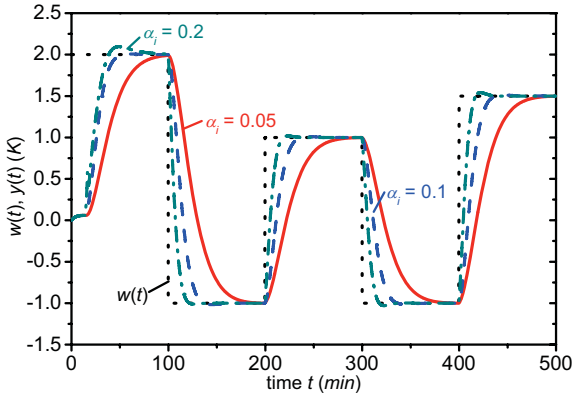


Fig. 3: Course of the reference signal $w(t)$ and the output variable $y(t)$ for various of the root α_i in adaptive control

Figures 3 and 4 represents simulation results for adaptive control. It can be clearly seen, that the increasing value of the parameter α_i results in the quicker output response, $y(t)$, but small overshoots. On the other hand, lower value of this root position is parsimonious to the input variable, $u(t)$, in Fig. 4 which could be in this case considered as a twist of the valve on the feeding of the cooling pipe.

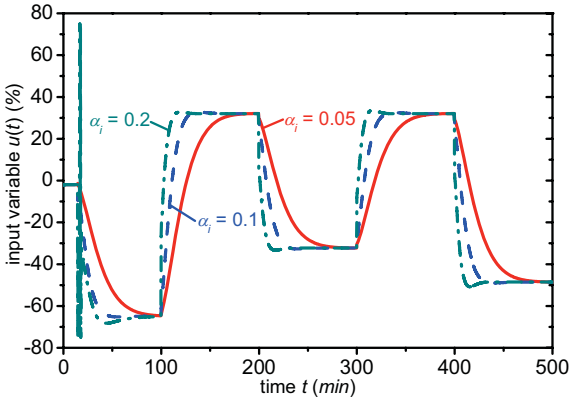


Fig. 4: Course of the input variable $u(t)$ for various of the root α_i in adaptive control

The quality of control was evaluated by the quality criteria S_u and S_y computed for a time interval as:

$$S_u = \sum_{i=2}^{N} \left(u(i) - u(i-1)\right)^2 [-]; \quad S_y = \sum_{i=1}^{N} \left(w(i) - y(i)\right)^2 [K^2] \quad \text{,for } N = \frac{T_f}{T_v}. \tag{36}$$

The results for both control strategies are shown in Table 2.

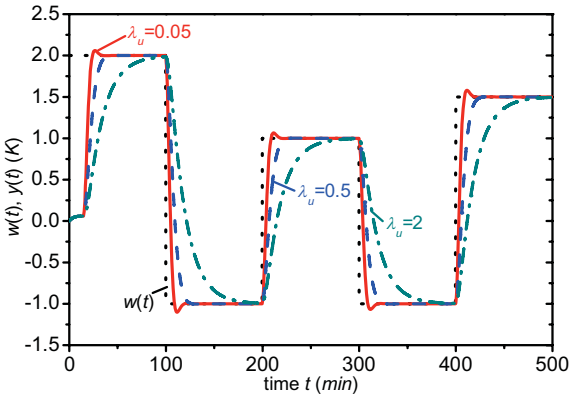


Fig. 5: Course of the reference signal $w(t)$ and the output variable $y(t)$ for various of the root λ_u in predictive control

The simulation results of the second control strategy, predictive control, are shown in Fig. 5 and Fig. 6. This controller is tuned via weighting parameter λ_u which is constant during the control and the second weighting parameter is $\delta_u = 1$ as it is written above in the theoretical part. In this case, increasing value of the λ_u results in slower response of the output variable $y(t)$.

The last two graphs in Fig. 7 and Fig. 8 compares the best results for both control strategies – adaptive control with $\alpha_i = 0.1$ and predictive control with $\lambda_u = 0.5$.We can say, that in this case both control strategies are comparable but the output response $y(t)$ in the predictive control reaches the reference signal (wanted value) $w(t)$ a little bit quicker than the adaptive controller.

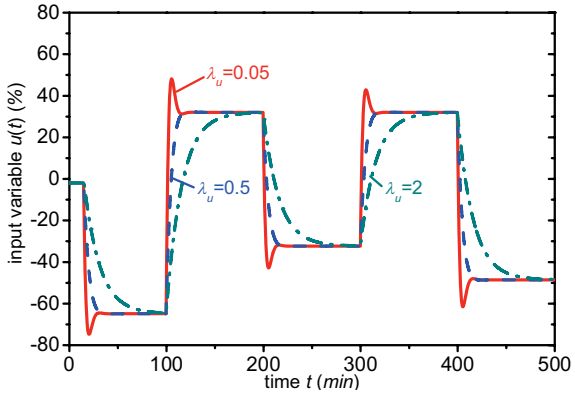


Fig. 6: Course of the input variable $u(t)$ for various of the root λ_u in predictive control

Table 2 The results of control quality criteria S_u and S_y

	Adaptive control, α_i =			**Predictive control, λ_u =**		
	0.05	0.1	0.4	0.05	0.5	2
S_u[-]	192.4	492.3	8571.5	3265.1	868.3	265.3
S_y[K^2]	1664.4	934.7	532.7	430.3	653.3	1194.8

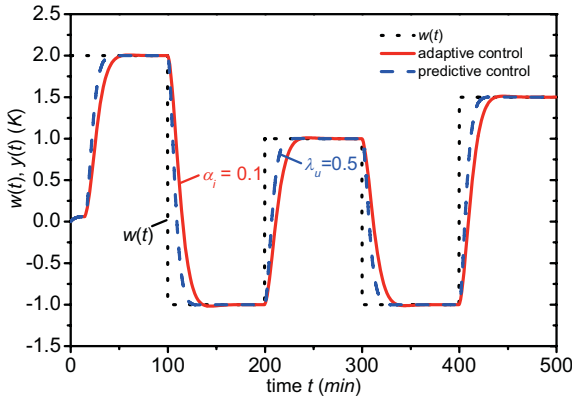


Fig. 7: The best courses of the output variable $y(t)$ for adaptive and predictive control

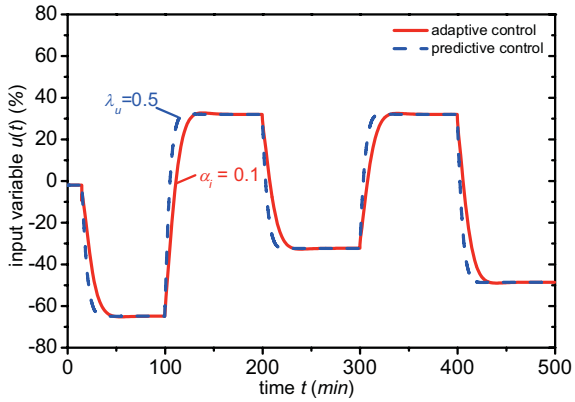


Fig. 8: The best courses of the input variable $u(t)$ for adaptive and predictive control

6. CONCLUSIONS

This paper presents two approaches which could be used for controlling of the temperature of the reactant inside the CSTR which is typical member of the nonlinear process with lumped parameters. Both, adaptive and predictive, controllers have good control results although the system has negative properties from the control point of view. The adaptive controller could be tuned by the parameter α_i while predictive controller has its weighting factor λ_u as a tuning parameter too. Final comparison of both control techniques results better for the predictive controller but the difference is minimal. The future work will be focused on the verification of the obtained results on the real plant which could increase reliability of these methods.

ACKNOWLEDGMENT

This work was supported by the Ministry of Education of the Czech Republic under grants No. MSM 7088352101 and No. 1M0567

REFERENCES

Bobál, V., Böhm, J., Fessl, J., Macháček, J. (2005) *Digital Self-tuning Controllers. Algorithms, Implementation and Applications*. Springer

Corriou, J.-P. (2004). *Process control. Theory and applications.* Springer-Verlag London.

Clarke, D.W., Mohtadi, C., Tuffs, P.S. (1987) Generalized predictive control – Part I. The basic algorithm. *Automatica.* 23(2), 137-148

Chen, H., Kremling, A., Allgöwer, F. (1995) Nonlinear Predictive Control of a Benchmark CSTR. *In: Proceedings of 3rd European Control Conference. Rome, Italy*

Fikar, M., Mikleš J. (1999) *System identification* (in Slovak). STU Bratislava.

Kučera, V. (1991) *Analysis and Design of Discrete Linear Control Systems*. Prentice-Hall, London

Kučera, V. (1993) Diophantine equations in control – A survey. *Automatica*, 29, 1361-1375

Kulhavý, R., Kárný, M. (1984) Tracking of slowly varying parameters by directional forgetting, *In: Proc. 9th IFAC World Congress*, vol. X, Budapest, p. 78-83

Middleton, R.H., Goodwin, G.C. (2004) *Digital Control and Estimation - A Unified Approach*. Prentice Hall, Englewood Cliffs.

Mukhopadhyay, S., Patra, A.G., Rao, G.P. (1992) New class of discrete-time models for continuos-time systeme. *International Journal of Control*, vol.55, 1161-1187

Stericker, D.L., Sinha, N.K. (1993) Identification of continuous-time systems from samples of input-output data using the δ-operator. *Control-Theory and Advanced Technology*, vol. 9, 113-125

Vojtesek, J., Dostal, P., Haber, R. (2004). Simulation and Control of a Continuous Stirred Tank Reactor. *In: Proc. of Sixth Portuguese Conference on Automatic Control CONTROLO 2004*. Faro. Portugal, p. 315-32

Gas-lift Optimization and Control with Nonlinear MPC

A. Plucenio * **D. J. Pagano** * **E. Camponogara** * **A. Traple** * **A. Teixeira** **

* *Departamento de Automação e Sistemas, Universidade Federal de Santa Catarina, 88040-900 Florianópolis-SC, Brazil*
e-mail: {plucenio, daniel, camponog, traple}@das.ufsc.br
** *CENPES-Petrobras, Rio de Janeiro, RJ, Brazil*
e-mail: alex.teixeira@petrobras.com.br

Abstract: More than 70% of the oil production in Brazil employs gas-lift as the artificial lift method. An effort is being done by some operators to complete new gas-lift wells with down hole pressure gages. This paper proposes a Non-Linear MPC algorithm to control a group of wells receiving gas from a common Gas-Lift Manifold. The objective is to maximize an economic function while minimizing the oscillations of the pressures at the manifold and at the bottom of the wells.

Keywords: Gas-Lift, Nonlinear MPC,

1. INTRODUCTION

Advanced Control Techniques like Nonlinear MPC have not arrived yet at the upstream processes of the oil industry. Many gas-lift wells with significant daily production are operating with manual driven gas injection and production chokes. In the last few years some Petroleum Exploration and Production companies have initiated efforts to introduce automation and control techniques in the operation of production wells. These initiatives resulted in technical approaches with names as smart wells, intelligent wells, smart fields or Digital Oilfield Management-GEDIG in Petrobras, Campos et al. (2006). The introduction of Information Technology in the oil production system is slow mainly due to the prohibitive cost of well intervention to install new sensors and actuators. Apart from that, sensors and actuators to be used in oil wells will have to cope with very harsh conditions caused by high pressure, temperatures and vibrations. There are several important works related to modeling, control and optimization of gas-lift wells operations like Boisard et al. (2002), Eikrem et al. (2004), L. Singre and Lemetayer (2006), Imsland et al. (2003), Camponogara and Nakashima (2006), Plucenio et al. (2006) to cite only a few. Not all control and optimization techniques discussed in the literature will be ready to be applied with the present instrumentation level of most gas-lift wells. This work discuss the automation of gas-lift wells equipped with downhole pressure measurement sensor, gas injection control valve and manually operated production choke. This is a realistic scenario in Brazil for new gas-lift wells. To our knowledge this is the first work that attempts to control the Gas Lift Manifold and the wells connected to it using Nonlinear MPC (NMPC). Section 2 discusses the NMPC formulation, section 3 presents and discusses the main results and section 4 concludes the paper.

⋆ A. Plucenio and A. Traple are supported by *Agência Nacional do Petróleo, Gás Natural e Biocombustiveis* under project PRH34-ANP/MCT. Daniel J. Pagano and E. Camponogara are supported by CENPES-Petrobras under the project *Development of control algorithms for artificial lifting*. A. Teixeira is a researcher at CENPES-Petrobras

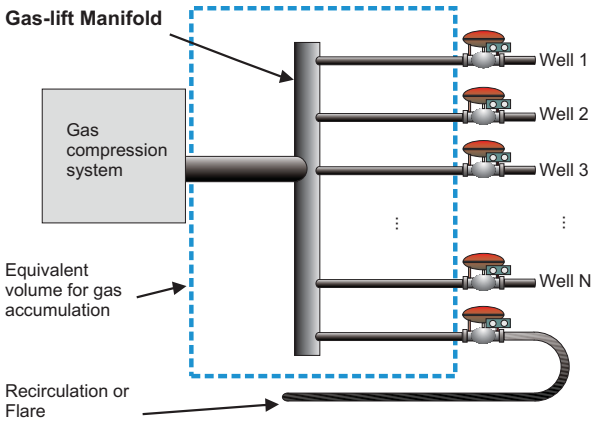


Fig. 1. Gas-lift Manifold

2. THE NMPC FORMULATION

We deal with a system where N gas-lifted wells with downhole pressure measurement and gas injection valves receive gas from a common Gas-lift Manifold (GLM). Gas from the compressor system enters the GLM and is distributed to the gas-lift wells and to an output which can be directed to the flare or to the recirculation of the compressor system. This output is a mechanism which allows gas to be discharged in cases where the gas flow-rate entering the GLM is higher than what is needed to operate the wells at their unconstrained optimum. This will be referred in the paper as the excess gas flow rate. The pressure at the GLM has to be kept at level high enough to allow injection in the annular of all gas-lift wells. For such a system shown in Figure 1 we wish

- to keep the GLM pressure close to a set-point designed according to the needs of the gas-lift wells,
- to distribute the gas flow-rate delivered by the compressor system among the gas lift wells in a way that maximizes an economic objective and

- to minimize the production oscillations caused by changes in the gas injection flow-rates. These oscillations cause problems to the separation process.

Some constraints should be introduced.

- To keep the gas injection flow rate of each well above a minimum value.
- To keep the pressure at the GLM between an upper and lower bound.

Table 1 presents the nomenclature used.

Table 1. Nomenclature

Symb.	Variable description	Unit
qo_i	Well i oil flow rate	std m^3.d^{-1}
$qliq_i$	Well i liquid flow rate	std m^3.d^{-1}
qw_i	Well i water flow rate	std m^3.d^{-1}
qg_i	Well i gas flow rate	std m^3.d^{-1}
p_{wf}	Bottom hole pressure with well flowing	kgf.cm^{-2}
$\bar{p}$	Average reservoir pressure	kgf.cm^{-2}
p_{sat}	Oil saturation pressure	kgf.cm^{-2}
q_{sat}	Liquid flow rate at $p_{wf} = P_{sat}$	std m^3.d^{-1}
q_{max}	Maximum well liquid flow rate (p$_{wf} = 0$)	std m^3.d^{-1}
qo_{max}	Maximum well oil flow rate	std m^3.d^{-1}
$qinj_i$	Well i gas injection flow rate	std m^3.d^{-1}
q_{exc}	GLM excess gas flow rate (flare or recirc.)	std m^3.d^{-1}
P_m	Gas Lift Manifold Pressure	kgf.cm^{-2}
P_{msp}	Gas Lift Manifold Pressure set point	kgf.cm^{-2}
p^*_{wf}	Value of p_{wf} used for normalization	kgf.cm^{-2}
$qinj^*$	Value of $qinj$ where $p_{wf} = p^*_{wf}$	std m^3.d^{-1}
q_{out}	Mass flow rate exiting the GLM	kgs^{-1}
q_{in}	Mass flow rate entering the GLM	kgs^{-1}
$\widetilde{x}$	Predicted or modeled value of x	

Symb.	Constants	Unit
V	Equivalent GLM volume	m^3
R	Universal Gas Constant, 8.314472	Pa.m$^3/Kmol$
M	Gas molecular weight	kg.mol^{-1}
BSW	Water saturation	-
GOR	Gas Oil Ratio	-

2.1 The NMPC Cost Function

The NMPC Cost Function should be tailored in such a way that its minimization provides the objectives discussed previously. There are several economic objectives that can be introduced in the Cost Function. A more general economic objective should express the net economic result of the gas lift operation taking into account the revenue from the oil production, gas production and the costs associated with the gas compression, water treatment, etc. Every well i has a maximum attainable oil production rate, qo_{maxi} , which can be obtained with an unique gas injection flow rate. Since there is a cost to implement the gas injection flow rate it becomes interesting to consider an economic objective which takes into account the gas compression cost and the revenue due to the oil produced. This is done at every sample time kTs computing the total amount of oil that will not be produced and the total amount of gas that will be injected between the actual time kTs and the future time T defined by the prediction horizon p and the sampling time Ts, $T = (k+p) * Ts$. An expression for the revenue loss due to oil production below unconstrained optimum is

$$L = P_o \sum_{i=1}^{N} \sum_{j=1}^{p} [qo_{maxi} - qo_i(k+j)]\, Ts, \text{ where} \quad (1)$$

P_o is the oil price per 1 $stdm^3$. The gas injection compression cost can be expressed as

$$C_{comp.} = C_c \sum_{i=1}^{N} \sum_{j=0}^{p-1} [qinj_i(k+j)]\, Ts, \text{ where} \quad (2)$$

C_c is the cost to compress $1stdm^3$ of gas to the GLM nominal pressure P_{msp}. The economic objective can be obtained by determining for every well i the vector

$$\Delta \mathbf{Q}inj_i = [\, \Delta qinj_i(k)\ \Delta qinj_i(k+1)\ \ldots\ \Delta qinj_i(k+m-1)\,]^T, \quad (3)$$

that minimize the objective function J_1, or,

$$\min_{\Delta \mathbf{Q}inj} J_1 \quad (4)$$

$$J_1 = \sum_{i=1}^{N} \sum_{j=1}^{p} [qo_{maxi} - qo_i(k+j)] + \frac{p}{m} \frac{C_c}{P_o} \sum_{i=1}^{N} \sum_{j=0}^{m-1} [qinj_i(k+j)]$$

The factor $\frac{p}{m}$ compensates the fact that the accumulated production loss is computed along an interval of time pTs while the total gas injected is computed along the time mTs where p and m are respectively the prediction and control horizon length. This formulation, in the absence of constraints is equivalent to the equal slope method, Kanu et al. (1981). One way to implement the cost function in matrix representation is to define for each well i the oil production loss qo_{Li},

$$qo_{Li} = qo_{maxi} - \tilde{qo}_i(p_{wfi}), \quad (5)$$

where $\tilde{qo}(p_{wf}))$ is computed with the predicted p_{wf}. We assemble the vector $\tilde{\mathbf{Qo}}_L$ with the difference between the maximum attainable oil flow rate and the predicted flow rate for every well i along the prediction horizon p.

$$\tilde{\mathbf{Qo}}L = [\, qo_{L1}(1)\ \cdots\ qo_{L1}(p)\ \cdots\ qo_{LN}(1)\ \ldots\ qo_{LN}(p)\,]^T \quad (6)$$

In order to damp the production oscillations we propose to minimize the sum of the time differential square of the production losses of all wells along the prediction horizon.

$$J_2 = \sum_{i=1}^{N} \sum_{j=1}^{p} \left(\frac{dqo_{Li}(k+j)}{dt} \right)^2 \quad (7)$$

The time differential is obtained using the matrix T equivalent to the $\Delta = 1 - z^{-1}$ operator.

$$T = \begin{bmatrix} -1 & 1 & 0 & 0 & \cdots & 0 \\ 0 & -1 & 1 & 0 & \cdots & 0 \\ \vdots & \vdots & \vdots & \vdots & \cdots & 0 \\ 0 & 0 & 0 & \cdots & -1 & 1 \end{bmatrix} \quad (8)$$

The final Cost Function used is

$$J = \mathbf{W_1}\tilde{\mathbf{Qo}}_L + \mathbf{W_2}\mathbf{Qm_{out}} + (\mathbf{P}_{msp} - \tilde{\mathbf{P}}_m)^T \mathbf{W}_3 (\mathbf{P}_{msp} - \tilde{\mathbf{P}}_m) + \left(\mathbf{T}\tilde{\mathbf{Qo}}_L\right)^T \mathbf{W}_4 \left(\mathbf{T}\tilde{\mathbf{Qo}}_L\right) + \Delta \mathbf{Qm}_{out}^T \mathbf{W}_5 \Delta \mathbf{Qm}_{out}, \text{where}$$
$$\mathbf{Qm}_{out} = [\, \mathbf{Q}inj_1;\ \mathbf{Q}inj_2;\ \cdots\ \mathbf{Q}inj_N;\ \mathbf{Q}_{exc.}\,], \text{ and}$$
$$\mathbf{Q}_{exc.} = [\, q_{exc}(k)\ q_{exc}(k+1)\ \ldots\ q_{exc}(k+m-1)\,]^T \quad (9)$$

The first two terms implement the economic objective, the third term forces the pressure at the GLM to its nominal value or set-point, the fourth term minimizes the production losses oscillation and consequently the oil production rate oscillations and the fifth term minimizes the changes in the gas injection flow-rates. The vectors $\mathbf{W}_1$ and $\mathbf{W}_2$ can be adjusted to implement the economic objective in steady state. The matrix $\mathbf{W}_3$, $\mathbf{W}_4$ and $\mathbf{W}_5$ must be adjusted to weight the dynamic response objectives against the economic objective. There is no doubt that the optimum gas distribution is reached in steady state since in this case terms 3, 4 and 5 vanish but a proper tuning should provide optimization also during production transients. An excessive requirement for the oil production oscillation attenuation may induce a production loss compared to softening this objective.

2.2 Prediction models

The main purpose of applying automatic control to a group of gas-lift wells is to maximize an economic objective. That means to distribute the available gas flow rate entering the GLM among the wells in order to maximize the oil production for instance. Using the available downhole pressure measurements, the parameters of the Inflow Performance Relationship (IPR) of each well as well as parameters like BSW and GOR, it is possible to estimate the oil production flow rate entering the well. For under-saturated reservoir (formation pressure above the bubble point pressure),

$$qo = J(\bar{p} - p_{wf}), \tag{10}$$

where J is the productivity index, p_{wf} is the well flowing pressure in front of the perforated zone, $\bar{p}$ is the static pressure, and qo is the oil flow rate produced by the well. For saturated reservoirs, Vogel's formula Vogel (1968) gives

$$qo = qv_{max}\left[1 - 0.2\frac{p_{wf}}{\bar{p}} - 0.8\left(\frac{p_{wf}}{\bar{p}}\right)^2\right], \tag{11}$$

where qv_{max} is the maximum oil flow rate (for $p_{wf} = 0$). Defining the bubble pressure as p_{sat}, Patton and Goland (1980) proposed an expression considering the case where $\bar{p} > p_{sat}$ and the well operating with $p_{wf} \geq p_{sat}$ or $p_{wf} < p_{sat}$:

if $p_{wf} \geq p_{sat}$

$$q_{liq} = \frac{q_{sat}}{\bar{p} - p_{sat}}(\bar{p} - p_{wf}), \tag{12}$$

if $p_{wf} < p_{sat}$

$$q_{liq} = q_{sat} + (q_{max} - q_{sat})\left[1 - 0.2\frac{p_{wf}}{p_{sat}} - 0.8\left(\frac{p_{wf}}{p_{sat}}\right)^2\right]$$

$$qw = BSW q_{liq.} \tag{13}$$

$$qo = (1 - BSW)q_{liq}, \tag{14}$$

$$qg = RGO q_o \tag{15}$$

where q_{liq} is an IPR relationship that accounts for liquid flow rate in saturated and under-saturated wells, qw is the water flow rate, qo is the oil flow rate and qg is the gas flow rate. Other IPR models are found in Fetkovich (1973), Richardson and Shaw (1982), Raghavan (1993), Wiggins et al. (1996), and Maravi (2003). Due to the difficulty to have on line measurements for oil, water and gas flow rate of each well, and taking advantage of the availability of downhole pressure measurements an effort was done to derive an empirical model relating steady state gas injection flow rate to downhole pressure. For a real application the cost to obtain steady state values of downhole pressure and gas injection rate is significant since the well will have to operate at downhole pressures which translate into lower oil flow rate. Therefore it is highly desirable that the steady state model relating $p_{wf} = f(q_{inj})$ could be adjusted with measurements close to the point (p^*_{wf}, q^*_{inj}) where the production loss is minimum. A mathematical model with good extrapolation capability is most welcome. Most gas-lift wells do not produce naturally and for those the knowledge of the average reservoir static pressure, even with some uncertainty, gives an important information that can be used in the model since $p_{wf}(q_{inj} = 0) = \bar{p}$. In order to avoid numerical problems the relationship proposed uses normalized variables. Downhole pressure and injection flow rate are normalized to the pair (p^*_{wf}, q^*_{inj}). This would be an operational point corresponding to an observed lowest downhole pressure. The exact point chosen to normalization is not too important as long as the curve adjustment can use points to the right and left of (p^*_{wf}, q^*_{inj}).

$$u = \frac{q_{inj.}}{q^*_{inj.}}$$

$$y = \frac{p_{wf}}{p^*_{wf}}$$

$$y = \Theta_1 e^{-\Theta_2 u^m} + \Theta_3 + \Theta_4 u^2$$

$$\tilde{p}_{wf} = p^*_{wf} y. \tag{16}$$

A simplified SQP algorithm was developed for the curve fitting

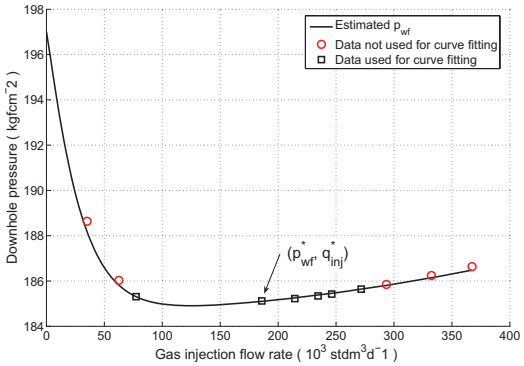


Fig. 2. Estimated p_{wf}

which uses the information about the average reservoir static pressure and its uncertainty. Figure 2 shows an example of curve fitting for data obtained from a rigorous steady state gas lift simulator. In order to verify the model extrapolation capability some points with lower values of downhole pressure were not used for the curve parameters adjustment. All points are plotted to show that the model adjusts well to the data even with a narrow data range used for the curve fitting. The points used for the curve fitting present downhole pressure close to the minimum which means that the production loss for obtaining these measurements would be minimum. Because one of the control objectives is to damp the oil production flow rate oscillations caused by changes in gas injection flow rates, it is also important to have a dynamic model for prediction. Since the main objective is economic, the dynamic prediction

model needs to exhibit a very accurate steady state relationship. Modern wells are being completed with Venturi gas-lift operating valves. These valves can provide critical flow for the injection gas at very low pressure drops (about 10% of the upstream pressure). This is normally enough to make sure that the gas-lift flow will be critical for most of its operating range. Critical flow in the gas-lift operating valve eliminates the heading phenomena but does not help to avoid the density wave oscillations, Hu (2004). In order to take advantage of the steady state model developed for $p_{wf} = f(q_{inj.})$ a Hammerstein model is proposed to be used for prediction. Figure

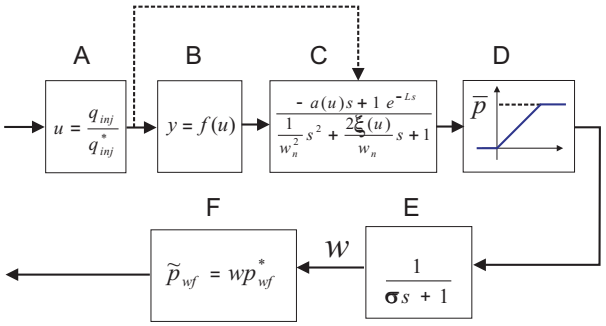


Fig. 3. Hammerstein dynamic model $p_{wf}(t) = f(q_{inj}(t))$

3 shows the dynamic model structure. Block A performs the normalization to the gas injection flow rate, block B applies the steady state function shown in equation (16), block C is a second order transfer function with transport delay and an adaptive zero and damping factor. Block D applies a saturation limiting the pressure values between zero and static pressure $\bar{p}$. Block E filters the saturation effect and block F multiplies the incoming signal by p^*_{wf} to recover the final estimated $\tilde{p}_{wf}$. It is assumed that the operating valve is working in critical mode so the gas flow rate crossing it is approximately the same gas flow rate that entered the casing head L seconds earlier. An approximate expression for L, ξ and the zero a are

$$L = \frac{H}{\sqrt{\frac{\gamma RT}{M}}},$$
$$\xi = k_1(.99 - e^{-3u(t-L)^2}) \text{ and}$$
$$a = \frac{k_2}{k_3 + u(t-L)}, \text{ where} \quad (17)$$

H is the distance from the casing head to the operating valve, γ is the gas ratio $\frac{c_p}{c_v}$, R is the universal gas constant, T is the gas temperature, $u(t-L)$ is the normalized gas injection flow rate at the casing head L seconds before the actual time and the constants k_1, k_2 and k_3 need to be tuned for each well together with the natural frequency w_n. In order to develop this empirical model several well cases were simulated with the OLGA$^{TM\,1}$. The pressure at the manifold, (P_m) is modeled as the pressure of a volume (V) filled with gas that results from the balance of gas that arrives from the compressor system and gas leaving to the wells and to the flare or to recirculation depending on the setup. The volume V is the sum of the internal volumes of all pipes between the compressor system and wells casing head. The pressure at the GLM is modeled as

[1] http://www.sptgroup.com/Products/olga/

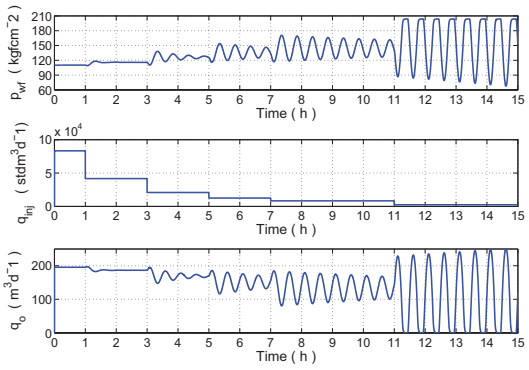


Fig. 4. Dynamic model response

$$\dot{P}_m = k_{GLM}(q_{in} - q_{out}),$$
$$k_{GLM} = \frac{RT}{MV}, \text{where} \quad (18)$$

T is the gas temperature, R is the gas universal constant and M is the gas molecular weight. The compressibility factor is assumed to be one. It is assumed that the gas mass flow rate entering the GLM is measured.

3. RESULTS AND DISCUSSION

In order to test the strategy proposed a total of 4 wells were modeled. These wells were simulated with a rigorous steady state simulator and the parameters of the empirical model given by equation (16) were tuned. The wells details are shown in table 2. A hypothetical dynamic model was added according to the model structure shown in figure 3. The Nonlinear MPC

Table 2. Well data

Parameter	*Well 1*	*Well 2*	*Well 3*	*Well 4*
$qmax[\frac{m^3}{d}]$	871.38	7.739e+003	5.177e+003	1.558e+003
BSW	0.341	0.676	0.03	0.488
$p^*_{wf}[kgf/cm^2]$	110.0	185.1	182.9	146.5
q^*_{inj} $[m^3/d]$	8.33e+4	1.859e+5	2.661e+5	9.98e+4
$\bar{p}$ $[kgf/cm^2]$	203.7	199	217.2	205
a_1	.9038	0.067	0.4003	0.3885
a_2	3.5039	8.0042	0.6133	5.8235
a_3	0.9666	0.9972	0.7751	0.9972
a_4	0.0075	0.0026	0.0082	0.0052
m	0.56	1.11	0.09	1.04

algorithm used is discussed in Plucenio et al. (2008b) and with more details in Plucenio et al. (2008a). The NMPC algorithm, named PNMPC, employs a continuous linearization technique where the vector of the predicted variable $\tilde{\mathbf{Y}}$ is represented by

$$\tilde{\mathbf{Y}} = \mathbf{F} + \mathbf{G}\Delta\mathbf{u} + \Gamma. \quad (19)$$

The matrix $\mathbf{G}$ is the Jacobian of $\tilde{\mathbf{Y}} = f(\Delta\mathbf{u})$ and is obtained by a numerical procedure realized in two steps at every iteration. The first step uses the matrix $\mathbf{G}$ of the previous iteration to produce an intermediate $\Delta\mathbf{u}$. Next, another matrix $\mathbf{G}$ is obtained using the $\Delta\mathbf{u}$ just computed. The matrix $\mathbf{G}$ used to compute the final $\Delta\mathbf{u}$ is an average of the two matrix. The vector Γ represents the correction factors which are an explicit version of the CARIMA error treatment.

3.1 NMPC Tuning

One way to tune the NMPC objective function weights is to start by the economic function as described by equation (4). Next the other weights are tuned to balance production oscillation attenuation with the economic objective. For numerical efficiency it may be interesting to multiply all weights by a common factor. The tuning parameters are shown in table 3. Controlling the system composed by the GLM and the wells

Table 3. NMPC Tuning Parameters

Symb.	Variable description	Value
Ts	Sampling time	5s
p	Prediction horizon for qo_L	150
p_1	Prediction horizon for P_m	18
w_1	Element of vector $\mathbf{W}_1$ $1 \times 4p$	.020
w_2	Element of vector $\mathbf{W}_2$ $1 \times 5m$	5e-4
w_3	Diagonal element of Matrix $\mathbf{W}_3$ $p_1 \times p_1$	(1)
w_4	Diagonal element of Matrix $\mathbf{W}_4$ $4p \times 4p$	(2)
w_5	Diagonal element of Matrix $\mathbf{W}_5$ $5m \times 5m$	(3)

(1) $w_3(i)$ varies linearly from 1 to 10 for $i = 1 : 18$
(2) w_4 is a linear function of the filtered and normalized gas mass flow rate q_{in} entering the GLM. 1 for $q^*_{in} = 1$ and 12 for $q^*_{in} = 0.25$
(3) $\mathbf{W}_5(i,i) = 1x10^{-5}$ for i=1:12. For Δu_{flare}, $\mathbf{W}_5(i,i)$ for i=13:15, a linear function of the filtered and normalized gas mass flow rate q_{in} entering the GLM was used. $\mathbf{W}_5(i,i)$ goes from $1x10^{-5}$ for $q^*_{in} = 0.25$ to $15x10^{-5}$ for $q^*_{in} = 0.25$.

has a great advantage of eliminating the gas lift availability constraint. Many gas-lift optimization studies consider gas lift availability as a constraint while this information is not always available. Another advantage is the possibility to apply optimization during the transients which can be more or less frequent depending on the setup used. On the other hand the GLM pressure dynamic behavior is highly dependent on the associated pipe internal volume and it will be normally faster than the downhole pressure. This requires the sampling time to be adjusted based on the GLM pressure dynamics. To overcome a bit the problem the qo_L predictions were done every 3 sampling time resulting in a prediction horizon p equal to 150. A constraint was used to make sure that the excess gas flow rate would be always positive. Besides, a minimum flow rate was imposed to all wells to avoid entering in the density wave limit cycle. A constraint was used to imposed a limit on the GLM Pressure deviation from the set point at +- 5%.

3.2 Results obtained

In order to test the control strategy proposed an operation of 24 hours was simulated covering different gas-lift availabilities. It was assumed an equivalent volume for the GLM (sum of all associated pipes internal volume) equal to 1 m^3. The initial gas injection flow rate was the sum of all gas flow rates values which corresponded to the values used for normalization. This value was considered as the nominal input GLM flow rate. Next the gas entering the GLM was changed to 50%, 25% and 110% of nominal value as shown on figure 5. Figure 6 top shows all the wells downhole pressure (normalized values) and the GLM pressure (normalized to the set-point value). It can be noticed that they change smoothly. The GLM pressure presents a small deviation from its set-point at moments of significant ramp type changes on the gas flow rate entering

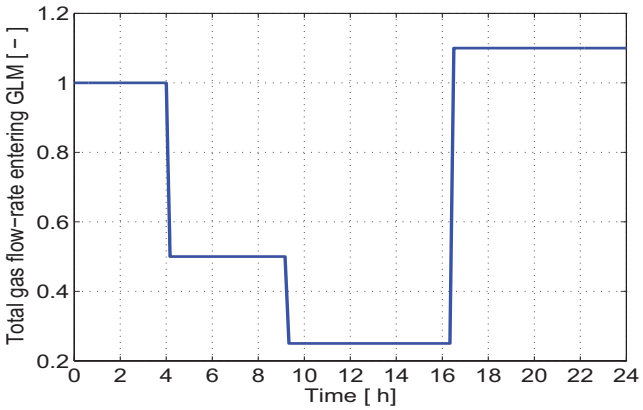


Fig. 5. Normalized gas flow-rate entering the GLM

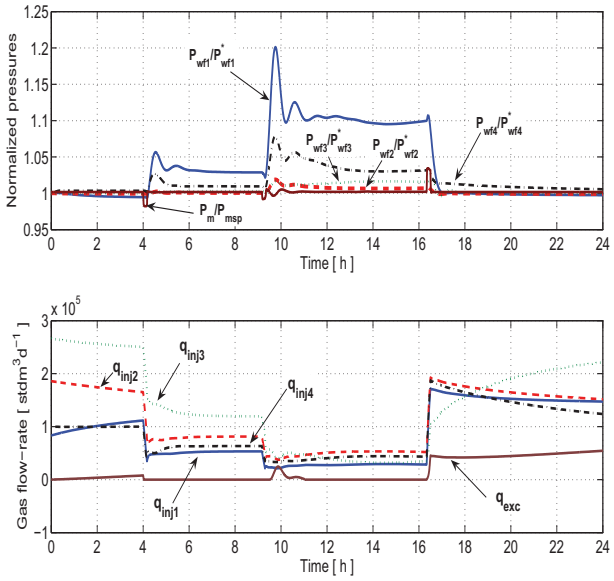


Fig. 6. GLM and gas lift wells behavior

the GLM although not enough to exceed the constraints. The bottom plot of figure 6 shows all the gas injection flow rates for the four wells and the excess gas flow rate. It is interesting to observe that when the gas flow rate entering the GLM goes to 110% of the nominal value the excess gas flow rate rises to keep the GLM pressure at its reference and to avoid production losses. When the gas entering the GLM decreases from 50% to 25% of the nominal value, the excess gas flow rate helped on avoiding too much change in the wells gas injection flow rate what would cause excessive oil production oscillation. This behavior can be controlled by tuning the Cost Function parameters. Figure 7 shows the evolution of the total oil production flow rate as the gas entering the GLM was changed. Both, 100% and 110% of nominal GLM input flow-rate give the same total oil production flow-rate. The gas flow rate not used for injection in the wells returns on the recirculation line as excess gas as shown in figure 6. The oil production flow rate of all the four wells is shown in figure 8. The solid lines were obtained with the simulation including the

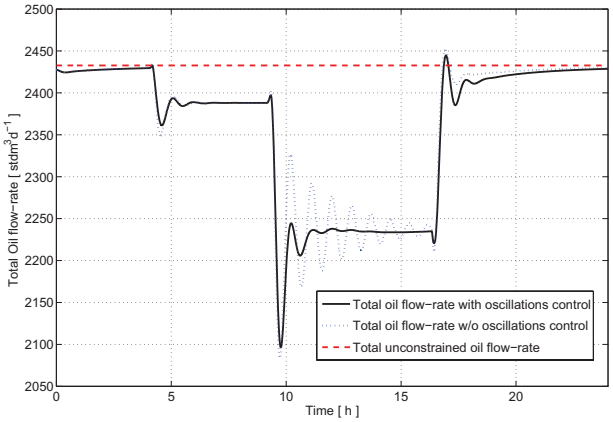


Fig. 7. Total oil production flow rate

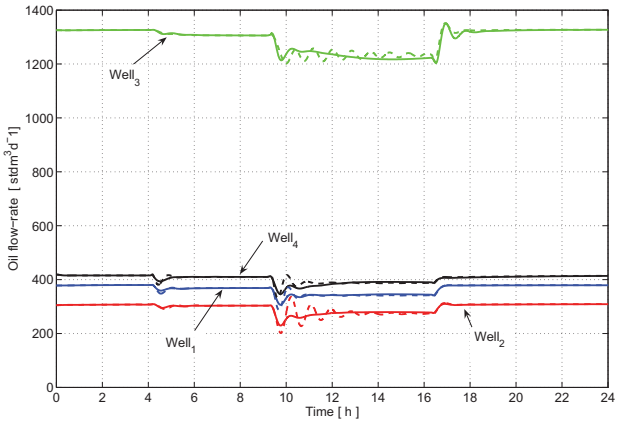


Fig. 8. Oil Production flow rate for all wells

oscillation attenuation control while the dashed lines not. The well 3 is the main producer. It is interesting to notice that the production decay is not much affected by the decrease in the total gas injection flow rate due to the appropriate gas allocation made by the NMPC algorithm. Despite the limited degree of freedom (only gas injection flow-rate manipulation) all the objectives are met; optimum gas distribution, GLM pressure control and attenuation of oil production oscillations.

4. CONCLUSION

Downhole permanent pressure measurement is becoming a reality for new wells. This work proposes the utilization of the PNMPC control technique discussed in Plucenio et al. (2008b) and demonstrates its applications on the control of 4 gas-lift wells using simulation. More work has to be done to investigate the quality of the empirical gas lift well dynamical model proposed.

ACKNOWLEDGEMENTS

The authors also acknowledge Scandpower for providing an academical OLGA2000 software license.

REFERENCES

Boisard, O., Makaya, B., Nzossi, A., Hamon, J., and Lemetayer, P. (2002). Automated well control increases performance of mature gas-lifted fields, Sendji case. In *Proc. 10^{th} Abu Dhabi International Petroleum Exhibition and Conference*. Abu Dhabi,. Paper SPE 78590.

Camponogara, E. and Nakashima, P.H.R. (2006). Optimizing gas-lift production of oil wells: Piecewise linear formulation and computational analysis. *IIE Transactions*, 38(2), 173–182.

Campos, S.R., Junior, M.F.S., Correa, J.F., Bolonhini, E.H., and Filho, D.F. (2006). Right time decision of artificial lift management for fast loop control. In *SPE Intelligent Energy Conference and Exhibition*. Amsterdam, Netherlands.

Eikrem, G.O., Imsland, L., and Foss, B. (2004). Stabilization of gas-lifted wells based on state estimation. *Proc. of International Symposium on Advanced Control of Chemical Processes-ADCHEM, Hong Kong*.

Fetkovich, M. (1973). The isochronal testing of oil wells. In *Proceedings of the SPE Annual Fall Meeting*. Las Vegas, Nevada.

Hu, B. (2004). *Characterizing gas-lift instabilities*. Master's thesis, Department of Petroleum Engineering and Applied Geophysics, Norwegian University of Science and Technology University, Trondheim, Norway.

Imsland, L., Eikrem, G.O., and Foss, B. (2003). State feedback control of a class of positive systems: Application to gas lift control. *Proc. of European Control Conference, Cambridge*.

Kanu, E.P., Mach, J., and Brown, K.E. (1981). Economic approach to oil production and gas allocation in continuous gas lift. *Journal of Petroleum Technology*, 33, 1887–1892. Paper SPE 9084.

L. Singre, N. Petit, T.S.P. and Lemetayer, P. (2006). Active control strategy for density-wave in gas-lifted wells. *International Symposium on Advanced Control of Chemical Processes-ADCHEM*, 2(ADCHEM 2006), 1075–1080.

Maravi, Y.D.C. (2003). *New Inflow Performance Relationships for Gas Condensate Reservoirs*. Master's thesis, Department of Petroleum Engineering, Texas A&M University.

Patton, D. and Goland, M. (1980). Generalized IPR curves for predicting well behavior. *Petroleum Engineering International*, 52(7), 74–82.

Plucenio, A., Mafra, G.A., and Pagano, D.J. (2006). A control strategy for an oil well operating via gas-lift. *International Symposium on Advanced Control of Chemical Processes-ADCHEM*, 2(ADCHEM 2006), 1081–1086.

Plucenio, A., Normey-Rico, J.E., Pagano, D.J., and Bruciapaglia, A.H. (2008a). Controle preditivo não linear na indústria do petróleo e gás. *IV Congresso Brasileiro de P & D em Petróleo e Gás - PDPetro*.

Plucenio, A., Pagano, D.J., Bruciapaglia, A.H., and Normey-Rico, J.E. (2008b). A pratical approach to predictive control for nonlinear processes. *NOLCOS 2008*.

Raghavan, R. (1993). *Well Test Analysis*. Prentice Hall, Englewood Cliffs, NJ.

Richardson, J.M. and Shaw, A.H. (1982). Two-rate IPR testing – a practical production tool. *Journal of Canadian Petroleum Technology*, 57–61.

Vogel, J.V. (1968). Inflow Performance Relationships for Solution-Gas Drive Wells. In *JPT*.

Wiggins, M.L., Russel, J.E., and Jennings, J. (1996). Analytical development of vogel-type inflow performance relationships. *SPE Journal*, 355–362.

Application of a New Scheme for Adaptive Unfalsified Control to a CSTR with Noisy Measurements ⋆

Tanet Wonghong and Sebastian Engell *

* *Process Dynamics and Operations Group, Department of Biochemical and Chemical Engineering, Technische Universität Dortmund, 44221 Dortmund, Germany*
(e-mail: {t.wonghong, s.engell}@bci.tu-dortmund.de)

Abstract:
In this paper, a new scheme for adaptive unfalsified control with non-ideal measurements is presented and demonstrated for a well-known example of a nonlinear plant, the continuous stirred tank reactor (CSTR) with the van-der-Vusse reaction scheme. In our adaptive control algorithm, there are two adaptation mechanisms: 1. Switching of the active controller in a fixed set of candidate controllers by the ϵ-hysteresis switching algorithm. 2. Adaptation of the set of controllers performed by a population-based evolutionary algorithm. In this paper, the effect of measurement errors on the adaptive control scheme is investigated. The total least squares method is used to perform the deconvolution of noisy signals.

Keywords: Adaptive Control; Unfalsified Control; Nonlinear Control, Reactor Control, Measurement Error

1. INTRODUCTION

The adaptive unfalsified control scheme was initially introduced by Safonov et al. (1997). The basic idea is to switch among candidate controllers in a predefined set of controllers. This approach does not require a plant model but uses the observed plant input-output data while one controller is active to decide on the switching to the next active controller. Further developments by Wang et al. (2005) led to the concept of cost-detectability, the proposal of a cost-detectable cost function, and the ϵ-hysteresis switching algorithm. Stability of the adaptive system was proven in Wang et al. (2005) in the sense that if the set of controllers contains stabilizing controllers with satisfactory performance, the scheme will ultimately switch to one of them. In Engell et al. (2007), Manuelli et al. (2007) and Dehghani et al. (2007), it was pointed out that the scheme in Wang et al. (2005) cannot detect instability of controllers that are not in the loop and may temporarily switch to destabilizing controllers. For this reason, the cost function proposed in Wang et al. (2005) is not suitable for evaluating controllers that are not in the loop, and cannot be used to adapt the controllers in the set.

To resolve this problem, a new scheme of adaptive unfalsified control was proposed in Engell et al. (2007). The key point was the introduction of a new fictitious error signal that can be computed using the estimated sensitivity function obtained by deconvolution between the fictitious reference signal and the fictitious error signal. This new signal is used in a new cost function that can measure the true performance of non-active controllers correctly. Based upon this new cost function, an adaptation of the set of controllers using evolutionary algorithm was performed. The scheme was demonstrated to work for the well-known non-minimum phase CSTR example with undisturbed measurements in Wonghong and Engell (2008).

In this paper, we extend the approach in Engell et al. (2007) to the case of noisy measurements. In the next section, we first review the new concept of unfalsified control with the ϵ-hysteresis switching algorithm as a switching mechanism for the active controller and the adaptation of the set of candidate controllers using evolution strategies. Then we investigate the effect of noise added to the plant output signal on the observed plant input-output data, the fictitious reference signal, and the fictitious error signal. We introduce the total least squares technique to solve the noisy deconvolution problem. This leads to an estimation of the sensitivity function in the non-ideal situation. In section 5, we present simulation studies for the CSTR example. We show that the adaptation of the controller parameters can be performed successfully under non-ideal measurements.

2. A NEW SCHEME FOR ADAPTIVE UNFALSIFIED CONTROL

We consider a SISO adaptive unfalsified control system $\Sigma(P, \hat{K})$ mapping r into (u, y). The system is defined on $\Sigma(P, \hat{K}) : \mathcal{L}_{2e} \rightarrow \mathcal{L}_{2e}$. The scheme of an adaptive unfalsified control system $\Sigma(P, \hat{K})$ is shown in Fig. 1.

The disturbed unknown plant $P : \mathcal{U} \rightarrow \mathcal{Y}$ is defined by

$$\mathbf{P} = \{(u, y, w) \subset \mathcal{U} \times \mathcal{Y} \times \mathcal{W} \mid y = Pu + w\}. \quad (1)$$

⋆ This work was supported by the NRW Graduate School of Production Engineering and Logistics at Technische Universität Dortmund.

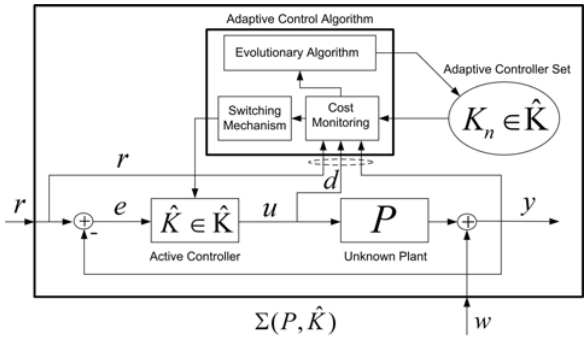


Fig. 1. Adaptive Unfalsified Control Scheme

The set of controllers $K : \mathcal{R} \times \mathcal{Y} \rightarrow \mathcal{U}$ is defined by

$$\hat{\mathbf{K}} = \{(r,u,y) \subset \mathcal{R} \times \mathcal{U} \times \mathcal{Y} \mid u = K_n \begin{bmatrix} r \\ y \end{bmatrix}\}, n = 1, 2, ..., N. \quad (2)$$

The signals $r(t), u(t), y(t)$ are assumed to be square-integrable over every bounded interval $[0, \tau], \tau \in \mathbb{R}_+$. The adaptive control algorithm maps vector signals $d = [r(t), u(t), y(t)]^T$ into a choice of a controller $K_n \in \hat{\mathbf{K}}$, where K_n satisfies the *stable causal left invertible* (SCLI) property [Wang et al. (2005)]. The true error signal is

$$e(t) = r(t) - y(t). \quad (3)$$

The adaptive control law has the form:

$$u(t) = \hat{K}(t) * e(t) \quad (4)$$

where $\hat{K} = K_{n(t)}$ denotes the active controller. $n(t)$ is a piecewise constant function with a finite number of switchings in any finite interval and $*$ denotes the convolution integral.

Let $d = (r(t), u(t), y(t)), 0 \leq t \leq T$ denote experimental plant data collected over the time interval T, and let $\mathbf{D}$ denote the set of all possible vector signals d. d_τ denotes the truncation of d, e.g., all past plant data up to current time τ. The data set $\mathbf{D}_\tau$ is defined by

$$\mathbf{D}_\tau = \{(r,u,y) \subset \mathcal{R} \times \mathcal{U} \times \mathcal{Y} \mid d_\tau = (r_\tau, u_\tau, y_\tau)\}.$$

We consider linear time-invariant control laws of the form:

$$K_n = \{(r,u,y) \subset \mathcal{R} \times \mathcal{U} \times \mathcal{Y} \mid u = c_n * e\} \quad (5)$$

where c_n is the impulse response of the n^{th} controller. $C_n(s)$ denotes the Laplace transform of c_n.

We assume that we have observed the excitation r_τ, the plant input data u_τ and the plant output data y_τ.

In unfalsified control, these data are used to evaluate whether the controller K_n meets a specified closed-loop performance criterion

$$J_n^*(r_\tau, u_\tau, y_\tau) \leq \alpha \quad (6)$$

where α is called the unfalsification threshold. If this condition is not met, the control law switches to a different controller and the previous controller is discarded. After at most N switchings, a suitable controller is found, if there is such a controller in the set.

The key idea of unfalsified control is to compute the cost $J_n^*(d_\tau, \tau)$ based upon the available measurements. For this purpose,

$$\tilde{r}_n = c_n^{-1} * u + y \quad (7)$$

and

$$\tilde{e}_n = \tilde{r}_n - y \quad (8)$$

are defined where c_n^{-1} is the impulse response of the inverse controller transfer function $C_n^{-1}(s)$. These signals are called the *fictitious reference signal* and the *fictitious error signal*, respectively. $\tilde{r}_n$ is the reference signal that produces the measured plant input u and output y if the controller K_n is in the loop instead of the currently active controller.

In Engell et al. (2007), the *new fictitious error signal*

$$e_n^* = \tilde{s}_n * r \quad (9)$$

where $\tilde{s}_n$ is the impulse response of the sensitivity function with the n^{th} controller in the loop,

$$\tilde{S}_n(s) = \frac{1}{1 + C_n P} \quad (10)$$

was introduced. e_n^* is the error that results for the true reference signal r with the controller K_n in the loop. As $\tilde{S}_n(s) = \frac{\tilde{E}_n(s)}{\tilde{R}_n(s)}$, $\tilde{s}_n$ can be computed via the deconvolution of $\tilde{r}_n$ and $\tilde{e}_n$ [Engell et al. (2007)].

When $J_n^*(e_{n_\tau}^*, d_\tau, \tau) > \alpha$, this implies that if the controller K_n were in the loop, it would not satisfy the performance criterion (6). In this case, it is said that the controller K_n is a *falsified* controller. Otherwise the controller K_n is an *unfalsified* controller.

The new adaptive control algorithm consists of two adaptation mechanisms:

1. Switching of the active controller
The closed-loop performances of all candidate controllers are computed using sampled signals

$$J_n^*(d_{\tau_k}, \tau_k) = \max_{\tau_j \leq \tau_k} \frac{\sum_{i=0}^{j} |e_n^*(i)|^2 + \gamma \cdot \sum_{i=0}^{j} |u(i)|^2}{\sum_{i=0}^{j} |r(i)|^2} \quad (11)$$

where γ is a positive constant.

The ϵ-hysteresis switching algorithm of Morse et al. (1992) is applied for the switching of the active controller:

(1) Initialize: Let $k = 0, \tau_0 = 0$; choose $\epsilon > 0$. Let $\hat{K}(0) = K_1, K_1 \in \hat{\mathbf{K}}(0)$, be the first active controller in the loop.
(2) $k = k + 1, \tau_k = \tau_{k+1}$
If $J^*(\hat{K}(k-1), d_{\tau_k}, \tau_k) \geq \min_{K_n \in \hat{\mathbf{K}}(k)} J_n^*(d_{\tau_k}, \tau_k) + \epsilon$,
then $\hat{K}(k) \leftarrow \arg\min_{K_n \in \hat{\mathbf{K}}(k)} J_n^*(d_{\tau_k}, \tau_k)$,
else $\hat{K}(k) \leftarrow \hat{K}(k-1)$.
(3) Go to 2.

2. Adaptation of the set of controllers $\hat{\mathbf{K}}(t^*)$
An evolutionary algorithm (EA) is used for the adaptation of the set of controllers because EA manipulate a population of candidate controllers and can handle nonconvex cost functions and are able to escape from local minima. The EA is executed only at units of time t^* after a sufficiently large change of $r(t)$ was detected. For accurate results, t^* should approximately match the settling time of the controlled system. Insufficient excitation leads to numerical problems due to an ill-conditioned matrix in the deconvolution. Thus, we restrict the activation of the EA to a suitable interval after a sufficient excitation by a change of $r(t)$. In this work, the evolutionary algorithm is a so-called *evolution strategy* where each individual is

represented by a vector of controller parameters and by a vector of strategy parameters that control the mutation strength.

The *evolution strategy* as introduced by Rechenberg (1965) and later developed by Schwefel (1975) is based on a population P of μ individuals $\mathbf{a} = (\mathbf{x}, \mathbf{s})$, which represent search points $\mathbf{x} = (x_1, \ldots, x_m) \in \mathbb{R}^m$ and vectors of strategy parameters $\mathbf{s} = (s_1, \ldots, s_m) \in \mathbb{R}_+^m$ that handle the evolution of the population. The size of the population is equal to the number of candidate controllers $\mu = N$. The μ parent individuals in the parent set are randomly selected from the population. The new offspring λ are generated by recombination of two parent individuals and by subsequent perturbation of single variable $x_j, j \in \{1, ..., m\}$ with a random number drawn from a Gaussian distribution $\mathcal{N}(0, s_j)$ by

$$x_j' = x_j + s_j \cdot \mathcal{N}(0, 1). \tag{12}$$

According to the self adaptation mechanism of evolution strategies, each strategy s_j is modified log-normally

$$s_j' = s_j \cdot \exp(\delta \cdot \mathcal{N}(0, 1)) \tag{13}$$

where δ is an external parameter. Normally it is inversely proportional to the square root of the problem size ($\delta \propto \frac{1}{\sqrt{N}}$). To preserve a constant number of individuals, the survivor selection chooses the μ best ($1 \leq \mu < \lambda = 7 \cdot \mu$) individuals out of the set of λ offspring ((μ, λ)-selection) or out of the union set of parents and offspring ($(\mu + \lambda)$-selection). The quality of each individual is evaluated by the fitness function $f(\mathbf{a}) = J^*(\mathbf{x}, d_{t^*}, t^*)$. As long as a fitness improvement ($\Delta f > min_{\Delta f}$) of the best individual within a certain number of generations can be observed, the termination criterion of the evolution strategy is not fulfilled and the μ selected individuals from the previous generation are used for the next iteration.

3. CONSIDERATION OF NOISE AT THE PLANT OUTPUT

For the scheme in Fig. 1 with a linear plant and a linear controller, in the Laplace domain,

$$\begin{aligned} Y(s) &= P(s)U(s) + W(s) \\ &= P(s)\hat{C}(s)(R(s) - Y(s)) + W(s) \\ &= \hat{T}(s)R(s) + \hat{S}(s)W(s) \\ &= Y_{true}(s) + Y_w(s) \end{aligned} \tag{14}$$

with the active complementary sensitivity function,

$$\hat{T}(s) = \frac{\hat{C}(s)P(s)}{1 + \hat{C}(s)P(s)} \tag{15}$$

and the active sensitivity function,

$$\hat{S}(s) = \frac{1}{1 + \hat{C}(s)P(s)}. \tag{16}$$

The observed disturbed plant input signal is,

$$\begin{aligned} U(s) &= \hat{C}(s)(R(s) - Y(s)) \\ &= \hat{C}(s)(R(s) - Y_{true}(s)) - \hat{C}(s)Y_w(s) \\ &= U_{true}(s) - \hat{C}(s)Y_w(s). \end{aligned} \tag{17}$$

Hence $y(t)$ and $u(t)$ consist of deterministic and stochastic components,

$$y(t) = y_{true}(t) + y_w(t) \tag{18}$$

$$u(t) = u_{true}(t) - \hat{c}(t) * y_w(t) \tag{19}$$

where

$$y_{true}(t) = \hat{t}(t) * r(t) \tag{20}$$

$$u_{true}(t) = \hat{c}(t) * (r(t) - y_{true}(t)) \tag{21}$$

$$y_w(t) = \hat{s}(t) * w(t). \tag{22}$$

Therefore, the error propagation in the measured plant input-output data depends on the closed-loop performance of the active controller.

4. STOCHASTIC DECONVOLUTION

4.1 Stochastic Fictitious Signals

Using (7), the stochastic fictitious reference signal $\tilde{R}_{i,w}$ of C_i using the noisy observed plant input-output data (U, Y) while controller $\hat{C}$ is active results as

$$\begin{aligned} \tilde{R}_{i,w} &= C_i^{-1}U + Y \\ &= C_i^{-1}\hat{C}(R - Y) + Y \\ &= C_i^{-1}\hat{C}(R - Y_{true} - Y_w) + Y_{true} + Y_w \\ &= \tilde{R}_{i,true} + \Delta\tilde{R}_i \end{aligned}$$

where $\Delta\tilde{R}_i = (1 - C_i^{-1}\hat{C})\hat{S}W$. Note that $\Delta\tilde{R}_{\hat{C}} = 0$.

Using (8), the stochastic fictitious error signal $\tilde{E}_{i,w}$ of C_i can be computed from (U, Y),

$$\begin{aligned} \tilde{E}_{i,w} &= C_i^{-1}U \\ &= C_i^{-1}\hat{C}(R - Y) \\ &= C_i^{-1}\hat{C}(R - Y_{true} - Y_w) \\ &= \tilde{E}_{i,true} + \Delta\tilde{E}_i \end{aligned}$$

where $\Delta\tilde{E}_i = -C_i^{-1}\hat{C}\hat{S}W$. Note that $\Delta\tilde{E}_{\hat{C}} = -Y_w \neq 0$.

Using (9), the new fictitious error signal $e_{i,w}^*(t)$ of C_i is

$$E_{i,w}^* = \tilde{S}_{i,w}R. \tag{23}$$

$\tilde{S}_{i,w}$ can be obtained using (10),

$$\begin{aligned} \tilde{E}_{i,w} &= \tilde{S}_{i,w}\tilde{R}_{i,w} \\ \tilde{e}_{i,w}(t) &= \tilde{s}_{i,w}(t) * \tilde{r}_{i,w}(t). \end{aligned} \tag{24}$$

From (24), $\tilde{s}_{i,w}(t)$ can be computed from $u(t)$ and $y(t)$ via $\tilde{e}_{i,w}(t)$ and $\tilde{r}_{i,w}(t)$. The noisy deconvolution is performed using sampled signals:

$$\begin{aligned} \tilde{\mathbf{R}}_{i,w} \cdot \tilde{\mathbf{s}}_{i,w} &= \tilde{\mathbf{e}}_{i,w} \\ (\tilde{\mathbf{R}}_{i,true} + \Delta\tilde{\mathbf{R}}_i) \cdot \tilde{\mathbf{s}}_{i,w} &= \tilde{\mathbf{e}}_{i,true} + \Delta\tilde{\mathbf{e}}_i \end{aligned} \tag{25}$$

where

$$\tilde{\mathbf{R}}_{i,true} = \begin{bmatrix} \tilde{r}_{i,true}(0) & 0 & \cdots & 0 \\ \vdots & \tilde{r}_{i,true}(0) & 0 & 0 \\ \tilde{r}_{i,true}(l-1) & \ddots & \tilde{r}_{i,true}(0) & 0 \\ \tilde{r}_{i,true}(l) & \tilde{r}_{i,true}(l-1) & \cdots & \tilde{r}_{i,true}(0) \end{bmatrix}$$

and $\tilde{\mathbf{e}}_{i,true} = [\tilde{e}_{i,true}(0) \ \cdots \ \tilde{e}_{i,true}(l-1) \ \tilde{e}_{i,true}(l)]^T$. The unknown matrix $\Delta\tilde{\mathbf{R}}_i$ is defined similar to $\tilde{\mathbf{R}}_{i,true}$ and the unknown vector $\Delta\tilde{\mathbf{e}}_i$ is defined similar to $\tilde{\mathbf{e}}_{i,true}$. Examples

of the computation of $\tilde{\mathbf{R}}_{i,true}, \tilde{\mathbf{e}}_{i,true}, \tilde{\mathbf{s}}_{i,true}$ can be seen in Engell et al. (2007) and Wonghong and Engell (2008).

The deconvolution problem (25) contains error terms both in the matrix and in the right hand side, and therefore it is not adequate to approach it as an ordinary least squares problem. Instead we employ the *total least squares* (TLS) method.

The total least squares method was originally proposed by Golub et al. (1980). The motivation comes from the asymmetry of the *least squares* (LS) method where no error term in the matrix $\tilde{\mathbf{R}}$ is taken into account. The idea of TLS is to find the minimal (in the Frobenius norm sense) error terms $\Delta\tilde{\mathbf{R}}$ and $\Delta\tilde{\mathbf{e}}$ in the matrix $\tilde{\mathbf{R}}$ and in the vector $\tilde{\mathbf{e}}$ that make the linear equations system (25) solvable, i.e.,

$$\{\tilde{\mathbf{s}}_{TLS}, \Delta\tilde{\mathbf{R}}_{TLS}, \Delta\tilde{\mathbf{e}}_{TLS}\} = \arg \min_{\tilde{\mathbf{s}}, \Delta\tilde{\mathbf{R}}, \Delta\tilde{\mathbf{e}}} \|\Delta\tilde{\mathbf{R}} \;\; \Delta\tilde{\mathbf{e}}\|_F$$

$$\text{subject to } (\tilde{\mathbf{R}} + \Delta\tilde{\mathbf{R}}) \cdot \tilde{\mathbf{s}} = \tilde{\mathbf{e}} + \Delta\tilde{\mathbf{e}}.$$

4.2 Solution of the Total Least Squares Problem

The conditions for the existence and the uniqueness of a TLS solution can be found in Markovsky et al. (2007):

$$\mathbf{Z} = [\tilde{\mathbf{R}} \;\; \tilde{\mathbf{e}}] = \mathbf{U}\Sigma\mathbf{V}^T$$

where $\Sigma = diag(\sigma_1, ..., \sigma_{l+1})$ is a singular value decomposition of $\mathbf{Z}$, $\sigma_1 \geq \cdots \geq \sigma_{l+1}$ are the singular values of $\mathbf{Z}$. Partitioned matrices are defined as

$$\mathbf{V} = \begin{bmatrix} \mathbf{V}_{11} & \vdots & \mathbf{v}_{12} \\ \cdots & & \cdots \\ \mathbf{v}_{21} & \vdots & v_{22} \end{bmatrix}, \Sigma = \begin{bmatrix} \Sigma_{11} & \vdots & \mathbf{0}_{12} \\ \cdots & & \cdots \\ \mathbf{0}_{21} & \vdots & \sigma_{l+1} \end{bmatrix},$$

where $\mathbf{V}_{11}, \Sigma_{11} = diag(\sigma_1, ..., \sigma_l) \in \mathbb{R}^{l\times l}, \mathbf{v}_{12}, \mathbf{0}_{12} \in \mathbb{R}^{l\times 1}, \mathbf{v}_{21}, \mathbf{0}_{21} \in \mathbb{R}^{1\times l}, v_{22}, \sigma_{l+1} \in \mathbb{R}$. A TLS solution exists if and only if v_{22} is not zero. In addition, it is unique if and only if $\sigma_l \neq \sigma_{l+1}$. In the case when a TLS solution exists and is unique, the solution is given by

$$\tilde{\mathbf{s}}_{TLS} = -\frac{\mathbf{v}_{12}}{v_{22}}. \tag{26}$$

Therefore, the new fictitious error signal $e^*_{i,w}(t)$ for controller C_i can be computed by

$$\mathbf{e}^*_{i,w} = \mathbf{R} \cdot \tilde{\mathbf{s}}_{i,w} = \mathbf{R} \cdot \tilde{\mathbf{s}}_{i_{TLS}}. \tag{27}$$

4.3 Ill-conditioned Matrix $\tilde{R}_{i,w}$

In the error-free case, the computation fails if $\tilde{\mathbf{R}}_{i,true}$ is ill-conditioned, in particular if $\tilde{r}_{i,true}(0) \to 0$. Using the relationship of $\tilde{r}_{i,true}(t)$ and $r(t)$

$$\tilde{R}_{i,true} = \frac{\hat{C}}{C_i} \frac{1 + C_i P}{1 + \hat{C} P} R \tag{28}$$

and applying the initial value theorem,

$$\tilde{r}_{i,true}(0) = \lim_{s\to\infty} s \frac{\hat{C}}{C_i} \frac{1 + C_i P}{1 + \hat{C} P} R. \tag{29}$$

If a unit step function is applied to $r(t)$ and P is strictly proper and $C_i(s) = k_{p_i}(1 + \frac{1}{T_{n_i} s})$,

$$\tilde{r}_{i,true}(0) = \frac{\hat{k}_p}{k_{p_i}}, \tag{30}$$

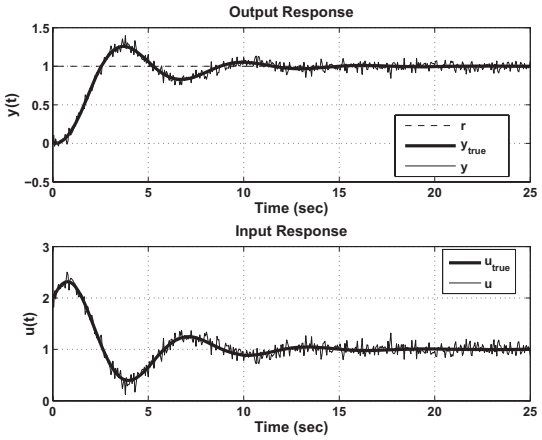


Fig. 2. The measured plant input-output data

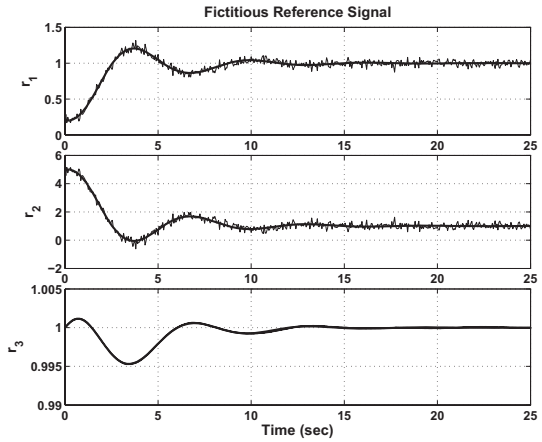


Fig. 3. Fictitious reference signals

so $|k_{p_i}| \gg |\hat{k}_p|$ leads to ill-conditioned matrices $\tilde{\mathbf{R}}_{i,true}$. For $\tilde{\mathbf{R}}_{i,w}$

$$\tilde{r}_{i,w}(0) = \tilde{r}_{i,true}(0) + \Delta\tilde{r}_i(0) = \frac{\hat{k}_p}{k_{p_i}} + \Delta\tilde{r}_i(0). \tag{31}$$

Hence the presence of measurement noise alleviates the ill-conditioning problem for the deconvolution technique.

4.4 Example of Estimated Fictitious Error Signals

We assume that $P = \frac{1}{(s+1)^3}$ and $\hat{C} = 2(1 + \frac{1}{3s})$ and a unit step was applied to $r(t)$. $d_\tau = (r_\tau, u_\tau, y_\tau)$ was observed up to $\tau = 25s$. We assume measurement errors in the plant output data as shown in Fig. 2. Three candidate controllers are tested: 1. $C_1 = 10(1+\frac{1}{3s})$ 2. $C_2 = 0.4(1+\frac{1}{3s})$ 3. $C_3 = 2(1 + \frac{1}{3s})$. $\tilde{r}_{i,true}(t)$ and $\tilde{r}_{i,w}(t)$ result as shown in Fig. 3. Note that the computation of $\tilde{r}_{3,w}(t)$ is error-free. $e^*_{i,true}(t)$ and $e^*_{i,w}(t)$ are shown in Fig. 4. The closed-loop instability of the loop with C_1 is detected and the performances of C_2 and C_3 are estimated well.

5. ADAPTIVE CONTROL OF A CSTR WITH NONMINIMUM PHASE BEHAVIOR

As an example of the application of the new adaptive control scheme to a nonlinear process we investigate the

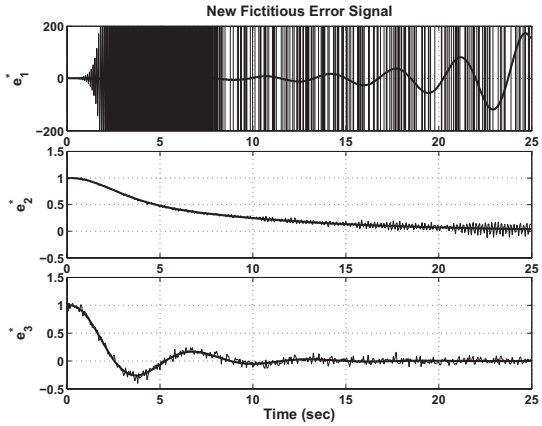


Fig. 4. New fictitious error signals

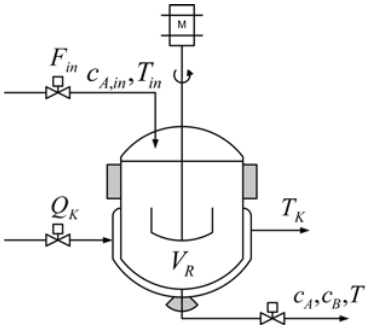


Fig. 5. Continuous Stirred Tank Reactor

well-known case study of the control of a CSTR with the van-der-Vusse reaction scheme. The parameters of the model are the same as in Engell et al. (1993), Klatt et al. (1998).

A sketch of the reactor is shown in Fig. 5. The reaction scheme is

$$A \overset{k_1}{\rightarrow} B \overset{k_2}{\rightarrow} C,$$
$$2A \overset{k_3}{\rightarrow} D.$$

The reactor is operated at a constant holdup, i.e., the volume of the contents is constant. The manipulated input $u(t)$ is the flow through the reactor, represented by the inverse of the residence time (F_{in}/V_R). u is in the range $0 \leq u(t) \leq 30h^{-1}$. We assume that the temperature control is tight so that the dependency of the kinetic parameters on the reactor temperature can be neglected. Under these assumptions, a SISO nonlinear model results from mass balances for the components A and B:

$$\begin{aligned} \dot{x}_1 &= -k_1 x_1 - k_3 x_1^2 + (x_{1,in} - x_1)u \\ \dot{x}_2 &= k_1 x_1 - k_2 x_2 - x_2 u \\ y &= x_2 \end{aligned} \tag{32}$$

where x_1 is the concentration of component A, x_2 is the concentration of component B and $x_{1,in}$ is the feed concentration of A, assumed to be constant. The parameter values are $k_1 = 15.0345h^{-1}, k_2 = 15.0345h^{-1}, k_3 = 2.324l \cdot mol^{-1} \cdot h^{-1}, x_{1,in} = 5.1mol \cdot l^{-1}$.

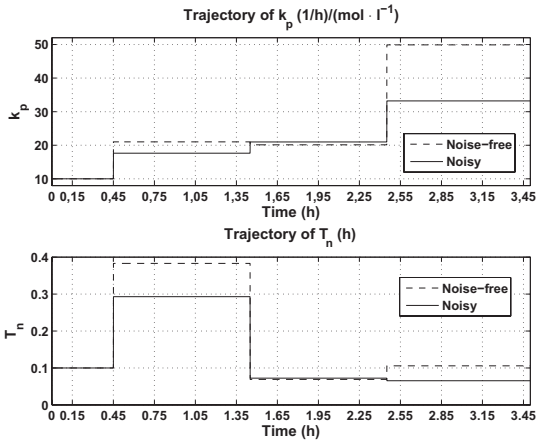


Fig. 6. Evolution of the active controller

We assume that the unknown plant P consists of the continuous stirred tank reactor as described by the above model plus a delay of $0.02h$ for the analytic instrument. The controller structure is a PI-controller defined by

$$C(s) = k_p(1 + \frac{1}{T_n s}).$$

The initial set of the controller parameters is given by the proportional gains $\mathbf{k}_{p_a} = \{10, 50, 100\}$ and the integral times $\mathbf{T}_{n_b} = \{0.1, 0.5, 1\}$. The initial set of candidate controllers consists of 9 candidate controllers with PI-controller parameter vectors $\Theta = \{\theta_i = [k_{p_a}, T_{n_b}]^T, 1 \leq a, b \leq 3\}$. The first active controller assigned to the feedback loop is $\theta_1 = [10, 0.1]^T \in \Theta$. All initial conditions at $\tau = 0$ are zero and the simulation horizon is $t_f = 3.5h$. The constant ϵ in the ϵ-hysteresis switching algorithm is 0.1 and $\gamma = 10^{-9}$ in the cost function J_i^*.

The reference signal is

$$r(t) = \begin{cases} 0mol \cdot l^{-1} & : \quad 0 \leq t < 0.15h; \\ 0.7mol \cdot l^{-1} & : \quad 0.15h \leq t < 1.15h; \\ 0.9mol \cdot l^{-1} & : \quad 1.15h \leq t < 2.15h; \\ 1.09mol \cdot l^{-1} & : \quad 2.15h \leq t < 3.5h. \end{cases}$$

EA activation times are at $t^* = 0.3h$ after each change of $r(t)$ at $t = 0.15h, 1.15h, 2.15h$.

CSTR with adaptation of the set of controllers with noisy measurements

The EA is executed three times at $0.45h$,$1.45h$, and $2.45h$. The EA used is a standard evolution strategy (ES) with adaptation of the search parameters according to Schwefel (1995) and Quagliarella et al. (1998). In this application, the size of the population is equal to the number of candidate controllers $\mu = N$. The $(\mu + \lambda)$ selection is chosen with $\mu = 9$ and $\lambda = 63$. This means that the best controllers are kept from the set of the old controllers and 63 offspring. The search space of solutions $\mathbf{k}_p \times \mathbf{T}_n$ is restricted to $[-100, 100] \times [0.01, 1]$ and the initial strategy parameters are set to 10% of the ranges of the variables. We assume Gaussian i.i.d. measurement errors and the TLS solution is used to compute the estimated sensitivity functions.

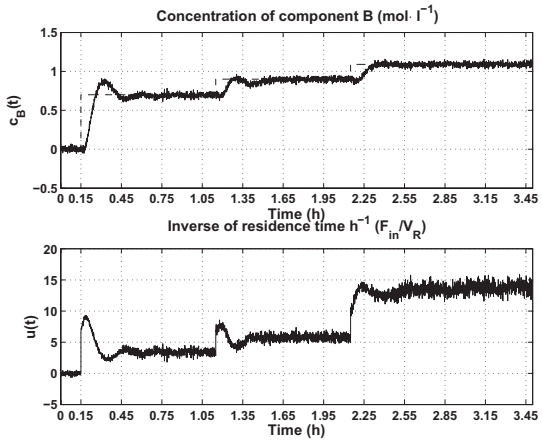


Fig. 7. Noisy control performance

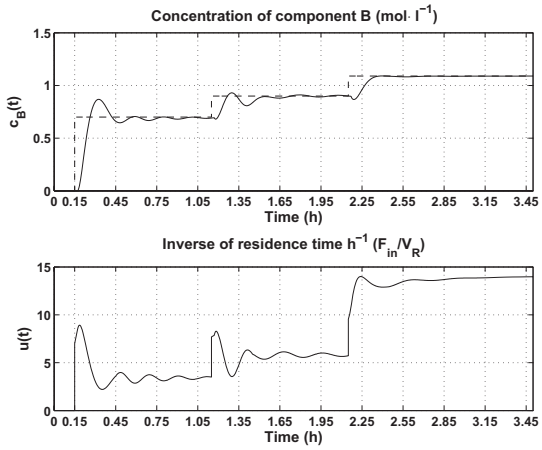


Fig. 8. Noise-free control performance

The first execution of the evolutionary algorithm was performed using measured data $d_{(0.15h,0.45h)}$ obtained with the first active controller θ_1 that was in the loop during $t \in (0.00h, 0.45h)$. At the first execution of the ES at $t = 0.45h$, the evolution strategy returns a new set of controllers for the first operating point after 38 generations. As shown in Fig. 6, the new active controller is $\theta^*_{p_{1_w}} = [17.6214, 0.2929]^T$.

The evolutionary algorithm was executed for the second time using the data $d_{(1.15h,1.45h)}$ with the active controller $\theta^*_{p_1}$. After 19 generations, the new active controller is $\theta^*_{p_{2_w}} = [20.9662, 0.0725]^T$ (see Fig. 6).

The evolutionary algorithm was executed for the third time using the data $d_{(2.15h,2.45h)}$ with the active controller $\theta^*_{p_2}$. After 12 generations the ES returned a new set of controllers and the new active controller is $\theta^*_{p_{3_w}} = [33.1989, 0.0652]^T$ (see Fig. 6). The control performance and the manipulated variable for the case with measurement noise are shown in Fig. 7 and can be compared with the noise-free case in Fig. 8. We can see that the active controller is well adapted to the change of the dynamics of the unknown plant under measurement error.

6. CONCLUSIONS

In this paper, the new scheme for adaptive unfalsified control was investigated for the case with noisy measurement. The deconvolution with noisy plant data can be solved by the total least squares method. The example of a CSTR with nonminimum phase nonlinear dynamics showed that a good performance can still be achieved for noisy measurements.

REFERENCES

A. Dehghani, B.D.O. Anderson and A. Lanzon. Unfalsified adaptive control: A new controller implementation and some remarks. *Proc. European Control Conf.*, Kos, 709-716, 2007.

S. Engell and K. U. Klatt. Nonlinear control of a non-minimum-phase CSTR. *Proc. American Control Conf.*, San Francisco, 2941 - 2945, 1993.

S. Engell, T. Tometzki and T. Wonghong. A new approach to adaptive unfalsified control. *Proc. European Control Conf.*, Kos, 1328-1333, 2007.

G. Golub and C. Van Loan An analysis of the total least squares problem. *SIAM J. Numer. Anal.*, 1980.

K.-U. Klatt and S. Engell. Gain-scheduling trajectory control of a continuous stirred tank reactor. *Computers & Chemical Engineering 22*, 491-502, 1998.

C. Manuelli, S.G. Cheong, E. Mosca and M.G. Safonov. Stability of unfalsified adaptive control with non $SCLI$ controllers and related performance under different prior knowledge. *Proc. European Control Conf.*, Kos, 702-708, 2007.

I. Markovsky and S. Van Huffel. Overview of total least squares methods. *Signal Processing*, 87, 2283-2302, 2007.

A.S. Morse and D.Q. Mayne and G.C. Goodwin. Application of Hysteresis Switching in Parameter Adaptive Control. *IEEE Transactions on Automatic Control*, 37:1343-1354, 1992.

D. Quagliarella, J. Periaux, C. Poloni and G. Winter. Genetic Algorithms and Evolution Strategy in Engineering and Computer Science: Recent Advances and Industrial Applications. *John Wiley & Sons*, 1998.

I. Rechenberg. Cybernetics Solution Path of an Experimental Problem. *Royal Aircraft Establishment Library Translation*, 1965.

M.G. Safonov and T.C. Tsao. The Unfalsified Control Concept and Learning. *IEEE Transactions on Automatic Control*, 42(6):843-847, 1997.

H.-P. Schwefel. Evoluionstrategie und Numerische Optimierung. *Dissertation*, 1975.

H.-P. Schwefel. Evolution and Optimum Seeking. *John Wiley & Sons, New York*, 1995.

R. Wang, A. Paul, M. Stefanovic and M.G. Safonov. Cost-detectability and Stability of Adaptive Control Systems. *Proc. IEEE CDC-ECC*, Seville, 3584-3589, 2005.

T. Wonghong and S. Engell. Application of a New Scheme for Adaptive Unfalsified Control to a CSTR. *Proc. IFAC World Congress*, Korea, 13247-13252, 2008.

Model Based Control of Large Scale Fed-Batch Baker's Yeast Fermentation

Akif Hocalar and Mustafa Türker

Pakmaya, P.K. 149, 41001 İzmit, Kocaeli, Türkiye
akifh@pakmaya.com.tr
mustafat@pakmaya.com.tr

Abstract : Two different control methods are applied to the technical scale (25 m^3) fed-batch baker's yeast fermentation. Feedback linearizing control design is used to manipulate the substrate feeding rate in order to maximize the biomass yield and minimizing the production of ethanol. Firstly, the specific growth rate controller is developed and applied to maintain the specific growth rate at specified trajectory. Secondly, the minimal ethanol controller is developed to maximize biomass productivity, by controlling specific growth rate just above the maximum oxidative growth rate by controlling ethanol concentration. The both controllers worked successfully and can be combined to follow required specific growth rate trajectory and respond successfully to disturbances in overflow fermentations such as *Saccharomyces cerevisiae.*

Keywords: Nonlinear control, feedback linearizing control, fed-batch, baker's yeast, specific growth rate, ethanol concentration, biocalorimetry.

1. INTRODUCTION

Fed-batch bioprocesses have extensive applications in industry for production of baker's yeast, enzymes, antibiotics, growth hormones, microbial cells, vitamins, amino acids and other organic acids (Perulekar and Lim, 1985; Yamane and Shimizu, 1984). *Saccharomyces cerevisiae* is used in many applications such as beverage products (beer, wine), baker's yeast for bread production, heterologous protein production, bio-transformations, flavour components, single cell protein, bio-ethanol, glycerol and food additives (Walker, 1998; Renard and Wouwer, 2008). The specific growth rate is a key variable for the growth-associated biotechnological processes and determines the physiological state of the cells and the capacity of cell's protein-synthesizing machinery that is important for recombinant protein production or biomass production in several fermentations (Cannizzaro et. al., 2004; Gnoth et. al., 2008). Similarly, *Escherichia coli* show similar metabolic behaviour in the presence of excess substrate and shortage of oxygen. In the production of recombinant proteins with *E. coli*, acetate is produced as an overflow metabolite both when *E.coli* grown under anaerobic or oxygen limiting conditions. It is important to maintain the specific growth rate below a certain threshold in order to avoid the accumulation of acetate throughout the fermentation (Rocha and Ferreira, 2002; Jana and Deb, 2005).

In the literature, many works have been reported by several authors for the control of fed-batch fermentation, most of these studies report experimental results either at laboratory scale or simulation results (Chen and Bastin, 1995; Pomerleau and Viel, 1992; Soons et al., 2006; Rocha and Ferreira, 2002; Cannizzaro et al., 2004; Valentinotti et al., 2003).

The main objective of this study is to develop a robust control scheme to cope with changing process dynamics during fermentation, set point tracking, required minimum process measurements and batch-to-batch consistency for the technical scale fed-batch baker's yeast fermentation. The control methods are based on previously developed and verified state estimation model and reliable measurement system (Hocalar et. al., 2006).

In this work, different key process variables are controlled with the state feed-back linearizing control scheme at technical scale fermentations: 1. nonlinear control of specific growth rate, 2. nonlinear control of ethanol concentration. In the first part, the derivation of nonlinear specific growth rate controller ant its results are discussed. The restrictive conditions of the controlling of specific growth rate are given at the end of first section. In the second part, the control of nonlinear ethanol concentration is presented. By means of controlling of overflow metabolite concentration at minimal concentration, specific growth rate can be maintained near the maximum values.

2. STOICHIOMETRY OF THE PROCESS

General stoichiometry of the baker's yeast fermentation process can be written with respect to reaction rates as (Türker, 2003; Türker, 2004):

$$r_s CH_2ON_{0.0125} + r_n NH_3 + r_o O_2 \rightarrow r_x CH_{1.8}O_{0.5}N_{0.2} + r_p CH_3O_{0.5} + r_c CO_2 + r_w H_2O + r_q \quad (1)$$

and the rate vector is

$$r = (-r_s\ ,\ -r_n\ ,-r_o\ ,\ r_x\ ,\ r_p\ ,\ r_c\ ,\ r_w\ ,\ r_q)^T \quad (2)$$

Metabolic heat production rate is added to reaction rates. The unknown process states are determined by the metabolic black-box modeling and the integration of estimated reaction rates. The redundant reaction rates are used in the derivation of reconciliated reaction rates *(Hocalar et. al., 2006)*.

3. RESULTS AND DISCUSSIONS

3.1. Nonlinear Control of Specific Growth Rate

The feedback linearizing control of the specific growth rate is based on the assumption of the presence of sufficient oxygen concentration and the absence of ethanol in the fermentation broth (Claes, 1999). In order to implement the control approach, the oxygen concentration has to be maintained high enough not to run into oxygen limitation throughout fermentation and the specific growth rate has to be below the critical value in order not to form ethanol.

The starting point for the derivation of the controller expression is the general dynamical mass balance equation for the substrate feeding as shown in Eq. 3.

$$\frac{dS}{dt} = D\,(S_{in} - S) - \left(\frac{\mu_x^{ox}}{Y_{X/S}^{ox}} + \frac{q_{e,pr}}{Y_{E/S}^{red}} + m \right) X \quad (3)$$

By rearranging the Eq. 3, Eq. 4 can be written as;

$$\frac{dC_s}{dt} = -\sigma X + \frac{F_s}{V}(C_{s,in} - C_s) \quad (4)$$

where $\sigma = \left(\frac{\mu_x^{ox}}{Y_{X/S}^{ox}} + \frac{q_{s,red}}{Y_{E/S}^{red}} + m \right)$, C_s is substrate concentration, X biomass, F substrate feed rate. The second step is to set up a stable linear reference model for tracking error. The reference model determines to the decreasing trajectory of the tracking error.

$$\frac{d}{dt}(C_s' - C_s) + \lambda\,(C_s' - C_s) = 0 \quad (5)$$

The λ is arbitrary adjustment coefficient and have to be chosen such that the differential equation (Eq. 5) is stable. At steady state conditions, the substrate concentration can be accepted zero, $(\frac{dC'}{dt} \approx 0)$ and Eq. 5 can be written as $\lambda\,(C_s' - C_s) = \frac{dC_s}{dt}$ and by substituting the Eq. 4 in the Eq. 5,

$$F_s = \frac{\sigma X - \lambda (C_{s,in} - C_s')}{C_{s,in} - C_s'} V \quad (6)$$

obtained as a final controller equation. Under oxidative conditions and in the absence of ethanol in the fermentation broth, specific growth rate is a function of substrate concentration. Therefore, specific growth rate (μ) can be written instead of substrate concentration term in Eq. 6. By rearranging the Eq. 6, the expression for substrate feed rate controller can be written as follows;

$$F_s = \frac{\frac{\mu_x}{Y_{X/S}^{ox}} X - \lambda\,(\mu_s - \mu')}{C_{s,in}} V \quad (7)$$

where λ is the arbitrary adjustment coefficient for the decrease of tracking error. When the Eq. 7 is applied to the fermentation, steady state errors are observed between the estimated specific growth rate and set profiles. In order to eliminate this difference, an integral term is added to the Eq. 7. and the obtained results are presented in Fig. 1 for the controlling of time varying specific growth rate profile.

$$F_s = \frac{\frac{\mu_x}{Y_{X/S}^{ox}} X - \lambda_P \left\{ (\mu_s - \mu') - \frac{1}{\lambda_i} \Sigma (\mu_s - \mu') \right\}}{C_{s,in}} V \quad (8)$$

The results of the implementation of Eq. 8 in a fed-batch fermentation are given Fig. 1. The adjustment parameters are $\lambda_p = 0.14$, $\lambda_i = 1800$ for the ascending and $\lambda_p = 0.27$, $\lambda_i = 1800$ for the descending specific growth rate region.

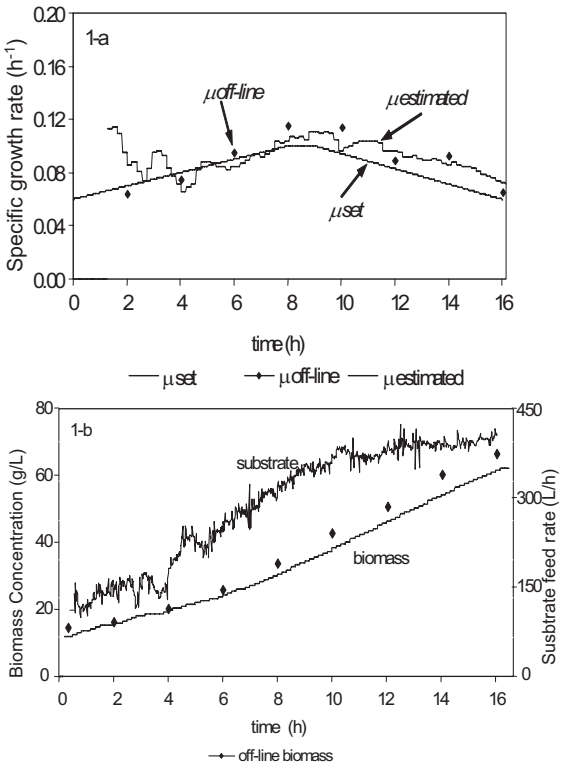


Figure 1.a- Specific growth rate b- biomass concentration and substrate.

The estimation of biomass concentration can be accepted successfully (Fig. 1-b) and is used in the calculation of specific growth rate (fig. 1-a). The specific growth rate estimations and off-line measurements are close to each other

with acceptable accuracy. The time varying specific growth rate profile is controlled successfully by the controller and obtained substrate feed rate resemble the predetermined substrate feeding profiles widely used in practice.

In Fig. 2, the results of different specific growth rate controlled fermentation are given. In this fermentation, ethanol formation is observed at the different times during the process and cause's decrease in the specific growth rate. The controller increased the substrate feed rate in order to compensate the decrease in the specific growth rate that caused more ethanol formation (Fig. 2-a). The unexpected decreases in the specific growth rate estimation are given in Fig. 2-b. The excess in the substrate feed rate puts the process more instability and results in failure of the control of specific growth rate. As a result, the substrate feed rate is manually intervened to consume the ethanol in the broth.

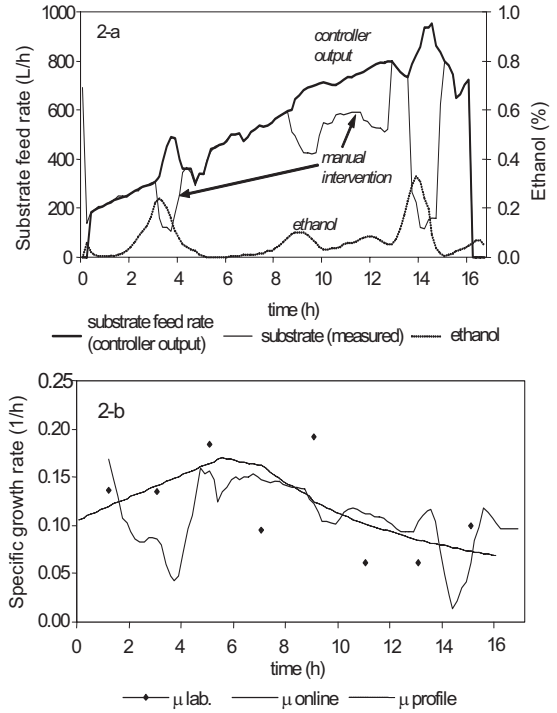


Figure 2.a- substrate feed rate and ethanol concentration, b-specific growth rate.

This controller has successfully controlled the specific growth rate at trajectory under defined conditions as shown in Fig. 1. Once the fermentation went to beyond the restrictive conditions the controller failed as shown in Fig. 2.

3.2. Nonlinear Control of Minimum Ethanol Concentration

An alternative way to control the specific growth rate at maximum oxidative rate is to use the overflow metabolite as an indicator of how close the actual value to critical growth rate to maximize biomass production. If ethanol concentration can be controlled at constant minimal concentration, it is possible to keep the specific growth rate slightly above the critical value (Cannizzaro et. al., 2004). In order to control the ethanol concentration, the regulator design is based on a feedback linearization of a reduced-order model of the process obtained by singular perturbation of the state space model under the following assumptions: the stoichiometric (yield) coefficients are known, the gaseous outflow rates (ethanol, CO_2, O_2) are measured on-line, the influent substrate concentration S_{in} is fixed and known, the specific growth rate is unknown. The singular perturbation techniques can be used for systems in which some reactions proceed at much faster rates than the others (Bastin and Dochain, 1990; Pomerleau and Viel, 1992; Chen et. al., 1995).

The dynamical process equations for five process states with known yield coefficients can be given as follows;

$$\frac{d}{dt}\begin{bmatrix} X \\ S \\ E \\ O \\ C \end{bmatrix} = \begin{bmatrix} 1 & 1 & 1 \\ -Y_{X/S}^{ox} & -Y_{X/S}^{red} & 0 \\ 0 & Y_{X/E}^{red} & -Y_{X/E}^{eth} \\ -Y_{X/O}^{ox} & 0 & -Y_{X/O}^{eth} \\ Y_{X/C}^{ox} & Y_{X/C}^{red} & Y_{X/C}^{eth} \end{bmatrix} \begin{bmatrix} \mu_x^{ox} \\ \mu_x^{red} \\ \mu_e^{ox} \end{bmatrix} X - D \begin{bmatrix} X \\ S \\ E \\ O \\ C \end{bmatrix} + \begin{bmatrix} 0 \\ DS_{in} \\ 0 \\ r_o \\ -r_c \end{bmatrix} \tag{9}$$

The general state space dynamical model can be written as follows (Bastin and Dochain, 1990);

$$\frac{d\xi}{dt} = K\varphi(\xi) - D\xi + F - Q \tag{10}$$

$$\xi^T = \left[X, S, E, O, C\right]$$

where ξ, Q, F involves n components, φ involves m reaction rates ve K is (NxM) size yield coefficient matrix. In Eq. 9, the first term $K\varphi(\xi)$ describes the kinetics of microbiological reactions, the remaining term $-D\xi + F - Q$ describes the transport dynamics of the components through the bioreactor. The yield coefficients used in the design is shown in Table 1.

Table 1. Parameters used in nonlinear controller design (Cmol/mol) (Beşli et. al. 1995).

k_1, $Y_{X/S}^{ox}$	3.65	k_2, $Y_{X/S}^{red}$	0.36
k_3, $Y_{X/E}^{red}$	0.19	k_4, $Y_{X/E}^{eth}$	1.35
k_5, $Y_{X/O}^{ox}$	1.56	k_6, $Y_{X/O}^{eth}$	0.83
k_7, $Y_{X/C}^{ox}$	1.45	k_8, $Y_{X/C}^{red}$	0.2
k_9, $Y_{X/C}^{eth}$	1.99		

By the systematic application of singular perturbation technique, fully reduced model can be established and in the case of $\dim(\xi_F) = M$ and K_F full rank, the process states can be partitioned as slow $\xi_S^T = \left[X, E\right]$ and fast varying state variables $\xi_F^T = \left[S, O, C\right]$. The substrate, oxygen and carbon-dioxide are fast varying state variables and biomass and ethanol are slow varying state variables for the fed-batch yeast fermentation process. The general dynamical model can be written as given in Eq. 11 by the assumption of the fast

varying state variables dynamics allow the singular perturbation (Bastin and Dochain, 1990);

$$\frac{d\xi_S}{dt} = K_S\,\varphi - D\,\xi_S + F_S - Q_S$$
$$K_F\,\varphi + F_F - Q_F = 0 \tag{11}$$

The reaction rate vector , $\varphi(\xi)$, can be written as;

$$\varphi(\xi) = -K_F^{-1}\,(F_F - Q_F) \tag{12}$$

By substituting Eq. 12 in Eq. 11, the dynamics of slow varying state variables can be obtained as in Eq. 13.

$$\frac{d}{dt}\begin{bmatrix} X \\ E \end{bmatrix} = -D\begin{bmatrix} X \\ E \end{bmatrix} + \begin{bmatrix} 1 & 1 & 1 \\ 0 & k_4 & -k_3 \end{bmatrix} * inv(K_F)\begin{bmatrix} DS_{in} \\ r_o \\ -r_c \end{bmatrix} \tag{13}$$

In steady state, singular perturbation allows the fast varying state's dynamics to consider to equal to zero and unknown reaction rates can be determined by simple matrixes operations as shown below if the inverse of the yield coefficient matrices can be calculated (Hocalar, 2007).

$$\begin{bmatrix} \mu_s^{ox} \\ \mu_s^{red} \\ \mu_e^{ox} \end{bmatrix} = -K_F^{-1}\begin{bmatrix} DS_{in} \\ r_o \\ -r_c \end{bmatrix} \tag{14}$$

By inserting the Eq. 14 in Eq. 13, slow varying process states can be calculated by means of basic matrice operations:

$$\psi = det(K_F) = \left(k_7 k_6 k_2 - k_5 k_9 k_2 - k_1 k_8 k_6\right)$$
$$\upsilon_1 = \left(-k_4 k_1 k_6 + k_3 k_2 k_5\right)\psi^{-1}$$
$$\upsilon_2 = \left(-k_4 k_1 k_9 + k_3 k_2 k_7 - k_3 k_1 k_8\right)\psi^{-1}$$
$$\upsilon_3 = \left(k_4 k_5 k_9 - k_4 k_6 k_7 + k_3 k_5 k_8\right)\psi^{-1}$$
$$\frac{dE}{dt} = -D\,E + \upsilon_3\,DS_{in} + \upsilon_2\,r_o - \upsilon_1\,r_c \tag{15}$$

By inserting the Eq. 15 into the first order reference model equation in Eq. 16, the controller law for substrate feed rate can be obtained as in Eq. 17.

$$\frac{d}{dt}(y^* - y) + (\lambda_1 + \lambda_2 x)(y^* - y) = 0 \quad (\lambda_1, \lambda_2 > 0) \tag{16}$$

$$F_s = \frac{1}{\upsilon_3}\left\{\frac{dE^*}{dt} + (\lambda_1 + \lambda_2 \hat{X})(E_s - E) + DE + \upsilon_1 r_c - \upsilon_2 r_o\right\} \tag{17}$$

The tracking ethanol error is tried to minimize using λ adjustment parameters. Several fed-batch experiments were conducted in a 25 m^3 fermentor to validate the control strategy. The results are given in Fig. 3.

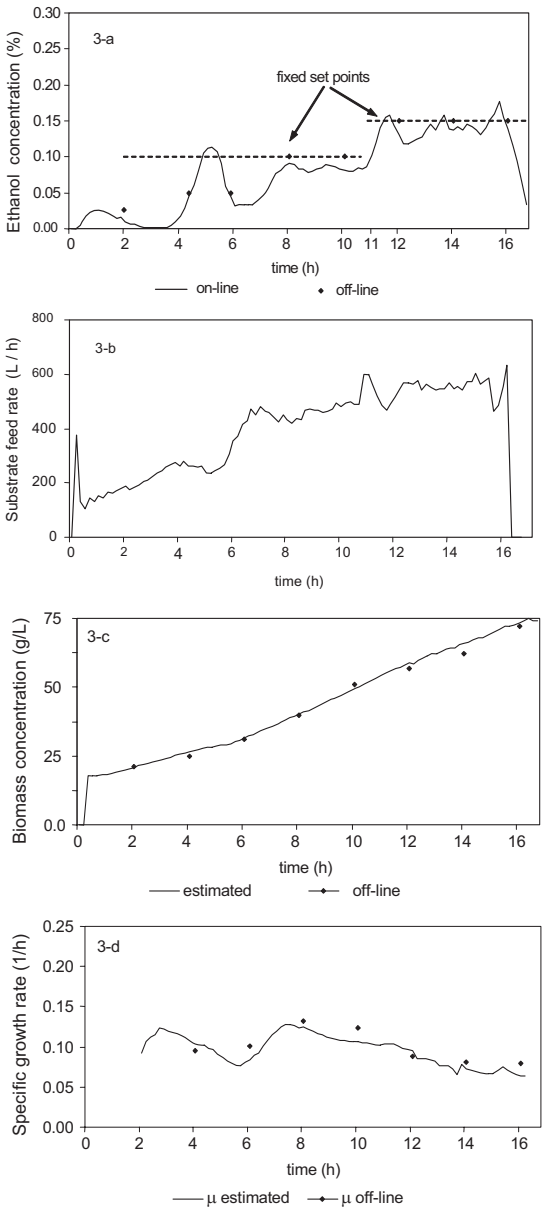


Figure 3: a- Ethanol concentration, b- substrate feed rate, c- biomass concentration and d- specific growth rate curves obtained from the industrial fermentation.

The controller was started at second hour and two fixed set points were tried to control for certain periods with ethanol set values E_s= % 0.10 and then with E_s= % 0.15. The ethanol concentration was successfully controlled at different set values from the 7th hour to the end of fermentation. The manipulated variable substrate feed rate and biomass concentration are given in Fig. 3-b and 3-c respectively. The biomass concentration increased exponentialy (Fig. 3.c) and the specific growth rate estimation is given in Fig. 3-d and quite close to experimental results. The controller developed stable response to the step change in the ethanol set point. The controller automatically adapted the feed rate of substrate to compensate for step changes. The difficulty of controlling the ethanol concentration can be seen in first

hours of fermentation (exponential growth phase). During the first hours, the controller increased the substrate feed rate and because of the time delay of the ethanol formation, slightly excess substrate feeding suggested by the controller.

4. CONCLUSION

The state feedback linearizing control strategy is applied to the industrial fed-batch baker's yeast fermentations. The control of specific growth rate and minimal ethanol concentration are attempted at technical scale fermentations. The control of specific growth rate at specificed trajectory is required in many fermentation processes. In this work, this approach has been successfully applied to baker's yeast fermentation. In order to maximize biomass concentration and productivity, the process has to be controlled at its maximum oxidative growth rate, minimizing by-product ethanol formation. This strategy is applied in second controller and specific growth rate was maintained slightly above maximum oxidative growth rate by maintaining and controlling by product ethanol at minimal concentration. This approach can also be applied to similar overflow processes such as the growth of *E. coli*. The ethanol concentration was controlled successfully at minimal concentrations. Both controllers can be combined to control specific growth rate at any trajectory and to minimize ethanol production.

NOMENCLATURE

C_i	concentration of i (kg/m^3)
D	dilution rate (1/h)
F_i	flow rate of i (m^3 / h)
K	yield coefficient matrices
M_i	molar weight of i (kg)
q_i	specific conversion rates of i (kg/kgh, C-mol/ C-mol h)
S	substrate concentration (kg/m^3)
X	biomass (kg/m^3)
V	volume (m^3)
$Y_{i/j}$	yield of i over j

Subscripts

ox	oxidative
red	reductive
eth	ethanol
m	maintenance
ae	aerobic
in	inlet
out	outlet
T	transpose
n	nitrogen
o	oxygen
e	ethanol
c	carbon
q	metabolic heat production
s	substrate
p	product
x	biomass
w	water

Greek Letters

μ	specific growth rate, (h^{-1})
ξ	state variable
λ	adjustment coefficient

REFERENCES

Bastin, G. and Dochain, D. (1990) On-line estimation and adaptive control of bioreactors. *Elsevier*.

Beşli, N., Türker, M., Gül, E. (1995). Design and simulation of a fuzzy controller for fedbatch yeast fermentation. *Bioprocess Engineering*. 13:141-148.

Cannizzaro, C., Valentinotti, S., von Stockar, U. (2004). Control of yeast fed-batch process through regulation of extracelluler ethanol metabolite. *Bioprocess Biosystem Engineering*. 26: 377-383.

Chen, L. (1992). Modelling, identifiability and control of complex biotechnological systems. *PhD Thesis*. Faculte des Sciences Appliquees. Universite Catholique de Louvan.

Chen, L., Bastin, G., Van Breusegem, V. (1995). A case study of adaptive nonlinear regulation of fed-batch biological reactors. *Automatica*. 31:1: 55-65.

Claes, J.E. (1999). Optimal adaptive control of the fed-batch bakers yeast fermentation process. *PhD Thesis*. Katholieke Universiteit Leuven.

Gnoth, S., Jenzsch, M., Simutis, R., Lubbert, A. (2008). Control of cultivation processes for recombinant protein production: a review. *Bioprocess and Biosystems Engineering*. 31: 21–39.

Hocalar, A. (2007). Model based control of industrial fed-batch baker's yeast fermentation process. *PhD Thesis*. University of Kocaeli, Turkey.

Hocalar, A., Türker, M., Öztürk, S. (2006). State estimation and error diagnosis in industrial fed-batch yeast fermentation. *AIChE Journal*. 52:11: 3967-3980.

Jana, S., and Deb, J. (2005). Strategies for efficient production of heterologous proteins in

Escherichia coli. *Applied Microbiology and Biotechnology*. 67: 289-298.

Perulekar, S.J. and Lim, H.C. (1985). Modeling, optimization and control of semi-batch bioreactors. *Advances in Biochemical Engineering*. 32: 207–258.

Pomerleau, Y., Viel, G. (1992). Industrial application of adaptive nonlinear control for baker's yeast production. *IFAC Modelling and Control of Biotechnological Processes*. Colorado, USA.

Renard, F. and Wouwer, A.V. (2008). Robust adaptive control of yeast fed-bacth cultures. *Computers and Chemical Engineering*. 32: 1246–1256.

Rocha, I. and Ferreira, E.C. (2002). Model-based adaptive control of acetate concentration during the production of recombinant protein with *E. Coli*. *IFAC 15th Triennial World Congress*. Barcelona, Spain.

Soons, Z.I.T.A., Voogt, J.A., Van Straten, G., Van Boxtel, A.J.B. (2006). Constant specific growth rate in fed-batch cultivation of *Bordetella Pertussis* using adaptive control. *Journal of Biotechnology*. 125: 252-268.

Türker, M. (2003). Measurement of metabolic heat in a production scale bioreactor by continuous and dynamic calorimetry. *Chemical Engineering Communication*. 190: 573-598.

Türker, M. (2004). Development of biocalorimetry as a technique for process monitoring and control in technical scale fermentations. *Thermochimica Acta*. 419: 73-81.

Valentinotti, S., Srinivasan, B., Holmberg, U., Bonvin, D., Cannizzaro, C., Rhiel, M., Von Stockar, U. (2003). Optimal operation of fed-batch fermentations via adaptive control of overflow metabolite. *Control Engineering Practice*. 11: 665-674.

Walker, G.M. (1998). *Yeast Physiology and Biotechnology*. Wiley. ISBN: 978-047196446-9.

Yamane, T. and Shimizu. S. (1984). Fed-batch techniques in microbial processes. *Advances in Biochemical Engineering*. 30: 148–194.

Modeling and Control of Free Radical Co-Polymerization

Harshad Ghodke, Siddarth Raman, B. Erik Ydstie

Department of Chemical Engineering, Carnegie Mellon University, Pittsburgh, USA (e-mail: ydstie@cmu.edu).

Abstract: In this work, we develop a model and a control system for a 4 monomer acrylate - methacrylate - styrene free-radical co-polymerization reaction. The model was implemented in PREDICI, and parameter estimation was carried out using nonlinear optimization from semi-batch experiments. Molecular weight distribution (MWD) determine utility. Stringent control over reactor conditions is critical. An inventory control scheme was demonstrated to work well for this complex polymerization process. A correlation mapping among the steady state initiator and monomer concentrations, molecular weight distributions and poly-dispersity was used to assign set-points for the inventory controller.

Keywords: polymerization, process control, passivity, automotive paint.

1. INTRODUCTION

We have developed a kinetic model for the free-radical polymerization of the Hydroxypropyl acrylate/ Styrene/ Butyl acylate/ Butyl methacrylate (HPA/ Sty/ BA/ BMA) copolymerization system. Unknown model parameters were estimated using data from semi-batch experiments and an inventory control strategy has been proposed for operation in continuous mode. The kinetic model has been implemented in PREDICI while the control system was implemented in MATLAB. The motivation of this current work stems from the need for effective operation of industrial continuous polymerization reactors to make resins from acrylates, methacrylates and styrene. These resins are an important constituent of automotive coatings.

Homopolymerization and butyl acrylate systems have previously been studied, and rate-constants are available in literature (Beuermann and Buback (2002), Maeder and Gilbert (1998), Curteanu (2003), Curteanu and Bulacovschi (2005)). For example, Congling Quan and Grady (2003) have developed a kinetic model of the high-temperature free-radical polymerization of butyl acrylate capable of predicting polymer molecular weight for a semi-batch process. D. Li and Hutchinson (2005) developed kinetic models for butyl acrylate - butyl methacrylate free radical polymerization system. In this study, the modeling tool PREDICI was used to show good predictions for the outputs, without a need for further refinement of the kinetic parameters. However, it is uncommon to find models with parameters that have been refined using a combination of experimentation and parameter optimization for this system.

The processability and utility of polymer products depend upon reactor operating conditions, and, good control is needed to achieve desired properties (Amrehn (1977), Elicabe and Meira (1988), MacGregor (1986)). Several promising control strategies applied to polymerization processes such as adaptive control (W. R. Cluett (1985)), optimal control (Choi (1997)), output feedback control (Soroush and Zambare (2000)) and nonlinear model predictive control (H. Seki (2001)) have been proposed. None of these approaches however, have been applied to very complex polymerization systems with several monomer inputs and few advanced schemes have found industrial application.

The purpose of our paper is to investigate the feasibility of using inventory control to control polymer properties. Inventory control is based on the idea of manipulating process flows so that the inventories follow their set points. The operator mapping flows to inventories in a macroscopic system is passive and any input strictly passive (ISP) feedback controller can be used in order to achieve input-output stability. The inventory control approach, which can be viewed as a way to chose candidate measured and manipulated variables for output linearization, was proposed by C. A. Farschman (1998). The method has been applied to transport reaction systems by M. Ruszkowski (2005). M. D. Díez (2007) applied the method to control particulate systems and Ydstie and Jiao (2006) applied the method to control a float glass plant for automotive windshield production. The HPA/ Sty/ BA/ BMA copolymerization system central to our work, has not been previously studied in literature from a modeling and control point of view. The model we propose combines literature data with experimental studies using nonlinear optimization.

2. KINETIC MODEL FOR FREE-RADICAL POLYMERIZATION

The model for free-radical polymerization consists of the following sets of reactions: Initiation, Propagation, Chain-transfer and Termination.

⋆ This work was supported by PPG Inc.

Initiation: The initiator used in this study is Di-*t*-amyl peroxide (DTAP) with a half life of 2 minutes at 160 °C (AkzoNobel (2006)). It decomposes with an efficiency f, to form two free radicals that can initiate propagation by reaction with any of the four monomers present in the system, abstract hydrogen atoms from other species in the system or recombine to form the initiator. Equation (1) represents the initiator decomposition reaction

$$I \xrightarrow{k_d} 2\,f\,I^{\cdot} \tag{1}$$

The dissociation rate constant k_d for DTAP is known to be given by eq (2) (AkzoNobel (2006))

$$k_d = 4.08 \times 10^{15} e^{-\frac{17831.36}{T}} \quad (s^{-1}) \tag{2}$$

The free radicals generated by initiator decomposition can initiate polymerization by reacting with the monomer species present in the solution. This process is described by the initiation reactions as shown in eq (3).

$$I^{\cdot} + M_i \xrightarrow{k_{p_{ii}}} P_1^{i\cdot} \tag{3}$$

Propagation: The polymer chain is then propagated by the products of the initiation reactions by reaction with monomers via the propagation reactions given in eq (4).

$$P_n^{i\cdot} + M_j \xrightarrow{k_{p_{ij}}} P_{n+1}^{j\cdot} \tag{4}$$

The rate constants for the homo-polymerization reactions, $k_{p_{ii}}$ (Asua (2007); D. Li and Hutchinson (2005)) were evaluated at 433 K and 10 psi. $k_{p_{11}}$ was assumed to be identical to $k_{p_{33}}$ because of lack of literature values. $k_{p_{33}}$ (and hence $k_{p_{11}}$) was assumed to be insensitive to pressure. Where possible, $k_{p_{ii}}$ were calculated using (6) and (5).

$$k_{p_{ii}} = A_i\, e^{-\frac{E_i + 1\times10^{-6}\Delta V_p\, P}{R\,T}} \qquad \forall i \in \{2,4\} \tag{5}$$

$$k_{p_{ii}} = A\, e^{-\frac{E_a}{R\,T}} \qquad \forall i \in \{1,3\} \tag{6}$$

The rate constants for the hetero-propagation reactions, $k_{p_{ij}}$, are obtained by defining reactivity ratios, r_{ij}, as shown in (7)

$$r_{ij} = \frac{k_{p_{ii}}}{k_{p_{ij}}} \tag{7}$$

Reactivity ratios were obtained from literature (Ham (1964); Chow (1975)). For the particular set of monomers present in the HPA/Sty/BA/BMA system, it was difficult to obtain reactivity ratios at the desired temperature. The values obtained from literature or calculated from Q-e data are available at temperatures different from the reactor temperature and hence, were treated as estimates.

Chain Transfer: Chain transfer in the context of free-radical polymerization involves the transfer of the radical from a live polymer chain to any other species present in the system which may be initiator, monomer, solvent molecules, dead polymer or live polymer or another species specifically added to the system which behaves as a chain transfer agent (CTA) (Asua (2007)). In this work, three types of chain transfer reactions are considered - chain transfer to monomer, chain transfer to solvent and chain transfer to live polymer.

Chain Transfer to Monomer: Chain transfer to monomers that contain aliphatic hydrogens such as acrylates and methacrylates involves H-atom abstraction to form an unsaturated radical (Asua (2007)). This unsaturated radical may then undergo further reaction as shown in eq (8). Transfer to monomer rates were not available in literature and were calculated using eq (9).

$$P_n^{i\cdot} + M_j \xrightarrow{k^{mon}_{tr_{ij}}} P_1^{j\cdot} + D_n \tag{8}$$

$$k^{mon}_{tr_{ij}} = \frac{k_{p_{ij}}}{C^{mon}_{tr}} \tag{9}$$

C^{mon}_{tr} is the coefficient for chain-transfer to monomer and typically lies in the range $1 - 50 \times 10^{-5}$ (Asua (2007)). For this work, the value C^{mon}_{tr} was taken to be 1×10^{-5}.

Chain-transfer to solvent: Chain transfer to solvent occurs according to reactions eq (10) and eq (11). In this system, for lack of literature values, we assume that both the solvents behave identically, and hence, have identical rate constants. The rate constants for chain transfer to solvent are obtained from eq (12) using a value for $C^{S_k}_{tr}$ is taken to be 3.16×10^{-5}, which is the logarithmic mean of the range in which it lies in 1×10^{-6} - 1×10^{-3}.

$$P_n^{i\cdot} + S_k \xrightarrow{k^{S_k}_{tr_{ii}}} S_k^{\cdot} + D_n \tag{10}$$

$$S_k^{\cdot} + M_i \xrightarrow{k^{S_k}_{ii}} P_1^{i\cdot} \tag{11}$$

$$k^{S_k}_{tr_{ii}} = \frac{k_{p_{ii}}}{C^{S_k}_{tr}} \tag{12}$$

Further, for lack of literature values, the rate constants $k^{S_k}_{ii}$, are assumed to be identical to the rate constant for propagation.

Chain transfer to Polymer: Chain transfer to polymer occurs according to reaction eq (13). In this system, for lack of literature values, we assume that the chain transfer to polymer is similar to chain transfer to monomer, and hence, has similar rate constants. The rate constants for chain transfer to polymer are obtained from eq (14), using a value of C^{pol}_{tr} equal to 1×10^{-5} which is the logarithmic mean of the range in which it lies 1×10^{-6} - 1×10^{-4}.

$$P_n^{i\cdot} + D_m \xrightarrow{k^{pol}_{tr_{ij}}} P_m^{j\cdot} + D_n \tag{13}$$

$$k^{pol}_{tr_{ij}} = \frac{k_{p_{ij}}}{C^{pol}_{tr}} \tag{14}$$

Termination: Termination occurs via two competing routes - combination (eqn (15)) and disproportionation (eqn (16)). Termination by combination results in the formation of a single dead polymer chain from two live polymer chains, whereas, termination by disproportionation results in the formation of one dead polymer chain and a live polymer chain with an unsaturated terminal residue which may react further.

$$P_n^{i\cdot} + P_r^{j\cdot} \xrightarrow{k_{tc_{ij}}} D_{n+r} \tag{15}$$

$$P_n^{i\cdot} + P_r^{j\cdot} \xrightarrow{k_{td_{ij}}} D_n + D_r \tag{16}$$

The termination rate constant is defined as the sum of the individual rate constants for termination by combination and disproportionation (eq (17)).

$$k_t = k_{tc} + k_{td} \tag{17}$$

Termination rates for free radical polymerization are diffusion controlled (Asua (2007)) and any available estimates are system specific. Thus, for the HPA/Sty/BA/BMA system, termination rates that were obtained from literature (Asua (2007)) were treated as estimates. The rate constant

for homo-termination for M_1 was not available in literature and was assumed to be identical to that of M_3.

The relative importance of the mechanism of termination by disproportionation versus termination by combination is measured using a parameter δ, which is defined in eq (18).

$$\delta = \frac{k_{td}}{k_{tc} + k_{td}} \tag{18}$$

δ_{ij} values for styrene and acrylates lie in the range of 0.05 - 0.2 while those for methacrylates lie in the range 0.5 - 0.8 (Asua (2007)). In this work, δ_{ii} for acrylates and styrene is taken to be 0.05, while that for methacrylates is taken as 0.65 (D. Li and Hutchinson (2005)). δ_{ij} value was evaluated as the arithmetic mean of δ_{ii} and δ_{jj}. The values for δ_{ii} were taken to be 0.05 for acrylates, 0.65 for methacrylates.

For the copolymerization reactions, $k_{tc_{ij}}$ and $k_{td_{ij}}$ are calculated using equations (19), (20) and (21). Equation (19) was obtained by generalizing eq (16) from D. Li and Hutchinson (2005) for the case of more than 2 monomers. Here, f_i represents the instantaneous mole fraction of M_i.

$$k_{t,copo_{ij}} = k_{t_{ii}}^{f_i}\, k_{t_{jj}}^{f_j} \tag{19}$$

$$k_{tc_{ij}} = (1 - \delta_{ij})\, k_{t,copo_{ij}} \tag{20}$$

$$k_{td_{ij}} = \delta_{ij}\, k_{t,copo_{ij}} \tag{21}$$

3. EXPERIMENTAL DATA AND PARAMETER ESTIMATION

Polymerization of the HPA/Sty/BA/BMA system was carried out in a well mixed, semi-batch reactor. The feed to the reactor consists of four monomers and an inhibitor dissolved in two solvents. The polymerization is carried out in semi-batch mode, in a 4000 mL vessel isothermally at a temperature of 160 °C and pressure of 10 Psi. The temperature of the vessel is maintained constant using electrical heating.

The reactor is initially charged with solvent mix (76 g S_1, 549.4 g S_2) and heated to 210 °C and then allowed to cool to 160 °C at atmospheric pressure. Feed A (293.6 g S_1 and 65 g I) and Feed B (827.2 g M_1, 393.9 g M_2, 374.2 g M_3 and 374.3 g M_4) are then fed to the reactor simultaneously for a period of 125 min and 120 min respectively. For the semi-batch process the outputs of interest are the residual monomer concentrations, weight average molecular weight (MW_w), number average molecular weight (MW_n), z-average molecular weight (MW_z) and the polydispersity index (PDI). Experimental data was obtained from two different sets of experiments. In one set, only data for the first 1000 seconds was collected and in the second set, data for times from 1000 - 7000 seconds was collected.

This model was implemented in PREDICI (Polyreaction Distributions by Countable System Integration), a comprehensive simulation package for the numerical integration of differential equations arising out of the kinetic equations describing polymerization systems. The results for the integration of the differential equations generated for the semi-batch process model in PREDICI are compared with the experimental data. The integration was performed in the *moment mode* with the values for the rate constants as listed in the tables and the relevant outputs were tracked. The model matches the experimental results poorly using literature data for the kinetic parameters.

Sensitivity and Estimability Analyses: The kinetic model for the HPA/Sty/BA/BMA system has a total of 79 parameters which can be estimated. To improve the fit of the model predictions and the experimental data, parameter estimation was carried out. The set of estimable parameters was obtained following the methodology of K. Zhen Yao (2003). Hence, as a first step toward identifying estimable parameters, a sensitivity study was carried out to identify the sensitivity of the model outputs with respect to the set of parameters as functions of time. The estimable parameters represent the set of parameters that affect the relevant set of outputs the most, among all the parameters, based on the initial values of the parameters at which the sensitivity derivatives are evaluated. The algorithm use to identify this set is called the Estimability analysis (K. Zhen Yao (2003)).

In order to be able to compare the measure of the sensitivity of a parameter, the sensitivity derivatives are non-dimensionlized using a scaling factor φ_{ijk} defined so that

$$\tilde{S}_{ijk} = \varphi_{ijk} \frac{\Delta y_{i,k}}{\Delta \theta_j}, \quad \varphi_{ijk} = \frac{\hat{\theta}_j}{\hat{y}_{i,k}} \tag{22}$$

$\hat{\theta}_j$ and $\hat{y}_{i,k}$ are used for scaling because they reflect the approximate magnitudes of changes in parameter estimates and model predictions respectively. Here, θ_j represents the j^{th} paramater and $y_{i,k}$ represents the value of the i^{th} output at the k^{th} time point.

It is important to note that the number of estimable parameters depends on the number of output variables, the number of observations per output variable and the linear dependence of the parameters on each other. Further, the global identifiability of the parameters depends on the size of the space of input-output variable values in which experimental data is available. For the purpose of estimability calculations, the scaled sensitivity derivatives defined in eq (**??**) were used to construct the *Scaled Sensitivity Matrix,* $\tilde{S}$ given in eq (23).

$$\tilde{S} = \begin{bmatrix} \varphi_{111}\dfrac{\partial y_{1,1}}{\partial \theta_1} & \cdots & \varphi_{1p1}\dfrac{\partial y_{1,1}}{\partial \theta_p} \\ \vdots & \ddots & \vdots \\ \varphi_{r11}\dfrac{\partial y_{r,1}}{\partial \theta_1} & \cdots & \varphi_{rp1}\dfrac{\partial y_{r,1}}{\partial \theta_p} \\ \varphi_{112}\dfrac{\partial y_{1,2}}{\partial \theta_1} & \cdots & \varphi_{1p2}\dfrac{\partial y_{1,2}}{\partial \theta_p} \\ \vdots & \ddots & \vdots \\ \varphi_{r1n}\dfrac{\partial y_{r,n}}{\partial \theta_1} & \cdots & \varphi_{rpn}\dfrac{\partial y_{r,n}}{\partial \theta_p} \end{bmatrix} \tag{23}$$

The cut-off (ϵ) value for which the algorithm is terminated is fairly arbitrary. From the estimability analysis for an $\epsilon = 1 \times 10^{-2}$, only 8 parameters may be estimated with $\mathcal{L} \equiv \{f, k_{t_{22}}, k_{p_{23}}, k_{p_{21}}, k_{p_{11}}, \delta_{22}, k_{p_{31}}, k_{p_{24}}\}$ in order of estimability.

Parameter Estimation: The parameter estimation algorithm was implemented by combining PREDICI and MATLAB using an MS Excel interface. The PREDICI-Excel link was set up to integrate the differential equations

based on initial conditions for parameters obtained from literature values. Updated estimates for the parameters were obtained from a weighted non-linear least squares Gauss - Newton algorithm with line search implemented in MATLAB. Table 1 shows the optimized values of the parameters.

Table 1. Optimized values of the estimable parameters

Parameter	Optimal value
f	0.95
$k_{t_{22}}$	1×10^9
k_{p23}	8483. 5
k_{p21}	426. 6
k_{p11}	3.5114×10^5
δ_{22}	0.2
k_{p31}	100
k_{p24}	2136

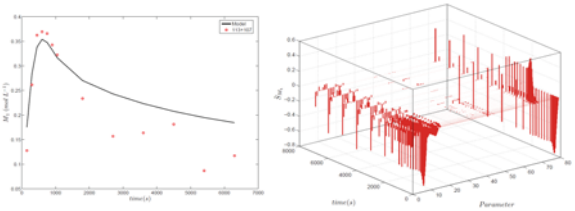


Fig. 1. Model results (solid) vs experiments (circles) for Monomer 1 (Left). Scaled sensitivity for Mononmer 1 shown with respect to time and the 79 parameters (Right).

Comparison plots shows that the model over-predicts some of the outputs and under-predicts some others. A typical result is shown in Figure 3.The sensitivity plot Figure 3 in shows that the sensitivity derivatives for the set of estimable parameters are of the same sign. This necessarily implies, that there is a trade-off between the outputs for which the model over-predicts and those for which the model under-predicts. This means that we can only obtain 'good' fits for either the set of outputs which are over-predicted or the set of outputs which are under-predicted. We conclude that the model we have developed misses some reaction mechanisms and that good fit with the semi-batch data is not possible.

4. CONTROL SYSTEM FOR THE COPOLYMERIZATION SYSTEM

Let the vector x represents the state of a process system, m a vector of control variables, d a vector of disturbance variables and y a vector of measurements. An inventory for the system described above is defined to be an additive continuous function v: $\mathbf{X} \rightarrow \Re$. For the system described above and using the nomenclature introduced in C. A. Farschman (1998) we have:

$$\frac{\mathrm{d}v}{\mathrm{d}t} = \phi(m, x, d) + p(x), \qquad v = g(z) \tag{24}$$

where $v \in \mathbb{R}_{+}^{dimv}$ are the inventories (mass, component mass,...). C. A. Farschman (1998) showed that the synthetic input and output pair

$$u = \phi + p + \frac{\mathrm{d}v^*}{\mathrm{d}t} \qquad e = (v - v^*) \tag{25}$$

is passive with the storage function $\psi = \frac{1}{2}(v-v^*)^T(v-v^*)$. C. A. Farschman (1998) implement a feedback-feedforward control in the form

$$u = -\mathbf{C}(e) = \phi(m, z, d) + p(z, d) + \frac{\mathrm{d}v^*}{\mathrm{d}t} \tag{26}$$

This control law is input strictly passive (ISP). Khalil (2002) showed that when a passive system is connected in feedback with an ISP controller, the closed loop is also passive. Hence the operator $\mathbf{C}$(e), which maps errors into synthetic controls, should be strictly passive. Most controllers in use are strictly passive. The control law is easy to implement if the inventories can be estimated from process data and the mapping $\phi(m, z, d)$ can be inverted with respect to the control variables m. In the simulations below we use a proportional controller ($\mathbf{C}(e) = Ke$ where K is a positive constant). It can be observed that an inventory controller linearizes the system dynamics.

In our application, the manipulated variables are the input flows of monomers (F_{M_1} to F_{M_4}) and initiator (F_I). The concentrations of the monomers and initiator are selected as the inventories and are forced to track their respective set-points. This gives a 5×5 multivariable control system for our system. Using the differential equation model (Equations 24 and 25), we generate the control equations for the required manipulated variables which force the concentration of the initiator and the respective monomers to its set-point. The control scheme generated is shown below: Control equation for initiator I,

$$F_{I_{in}} = F_{I_{out}} + k_d I V - K(I - I^*) \tag{27}$$

Control equations for the monomers,

$$F_{M_{1,in}} = F_{M_{1,out}} + \Big(2fk_dI + \sum_{j=1}^{4} P_n^{j\cdot} M_1(k_{p_{j1}} + k_{tr_{j1}} + \sum_{l=1}^{2} k_{11}^{S_l} S_l M_1)\Big)V - K(M_1 - M_1^*) \tag{28}$$

$$F_{M_{2,in}} = F_{M_{2,out}} + \Big(2fk_dI + \sum_{j=1}^{4} P_n^{j\cdot} M_2(k_{p_{j2}} + k_{tr_{j2}} + \sum_{l=1}^{2} k_{22}^{S_l} S_l M_2)\Big)V - K(M_2 - M_2^*) \tag{29}$$

$$F_{M_{3,in}} = F_{M_{3,out}} + \Big(2fk_dI + \sum_{j=1}^{4} P_n^{j\cdot} M_3(k_{p_{j3}} + k_{tr_{j3}} + \sum_{l=1}^{2} k_{33}^{S_l} S_l M_3)\Big)V - K(M_3 - M_3^*) \tag{30}$$

$$F_{M_{4,in}} = F_{M_{4,out}} + \Big(2fk_dI + \sum_{j=1}^{4} P_n^{j\cdot} M_4(k_{p_{j4}} + k_{tr_{j4}} + \sum_{l=1}^{2} k_{44}^{S_l} S_l M_4)\Big)V - K(M_4 - M_4^*) \tag{31}$$

In the above equations, K is a positive constant and I^* and $M_i^*(i \in \{1,2,3,4\})$ are the predetermined set-points for the initiator and monomer concentrations. The theory provides guidance for how to chose K since $1/K$ corresponds to the closed loop time-constant. The control

system assumes that the initiator and four monomer concentrations are measured. If these variables are not measured then it is necessary to develop an estimator to estimate these variables from the model.

To illustrate the performance of the control system we have introduced set point changes, first in M_1 at time 330 seconds and then in I at time 510 seconds to study how the system responds to these set point changes after the effect of initial conditions have died out. The results for the setpoint tracking performance for initiator and 3 monomers are shown in Figure 2 [1] . The proportional gain for all four controllers is $K = 0.01$ so that the closed loop time constant for all the four outputs is equal to 100 secs. The results show that the inventory controller decouples the response so that a setpoint change in one variable does not lead to a change in the other variables [2] . In practice there will be a mismatch between the real system and the controller equations and it may not be possible to achieve perfect decoupling.

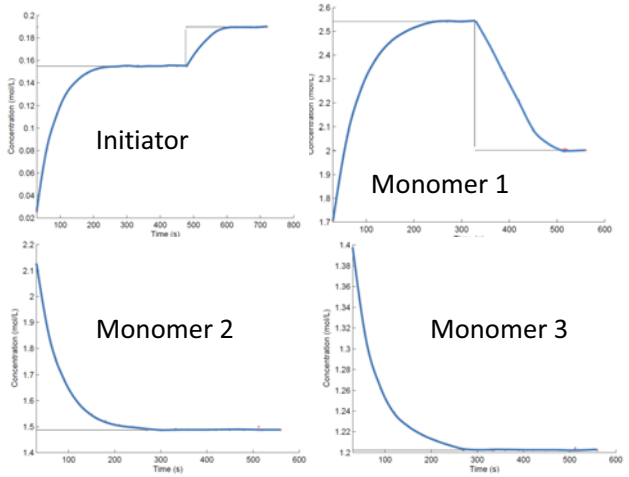


Fig. 2. Initiator, 3 Monomer Concentrations and Setpoints

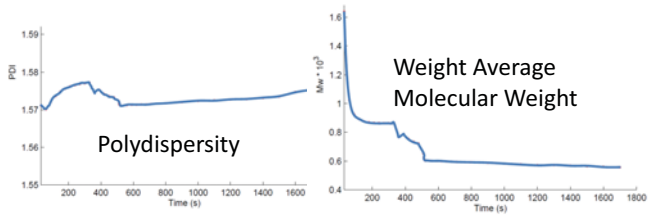


Fig. 3. Polydispersity and Weight Average Molecular Weight

5. THE CORRELATION MAPPING

The inventory control scheme allows the user to define setpoints in terms of inventories. In our case these are the initiator concentration and concentrations of monomers inside the reactor. These variables are related to the flow variables in a passive manner and they are easy to control. In practice it is often necessary to control secondary variables like molecular weight and polydispersity. These variables are more difficult to control since the relative degree of the control system now may be much higher. In this work we have solved the problem by developing a correlation mapping to identify how the steady state monomer set points correspond to target molecular weights and polydispersity for use in the inventory control scheme. The idea now is to use the mapping to generate inventory setpoint and control to the calculated setpoints. Model uncertainty can be compensated for using a separate estimation algorithm to fit the mapping to the data in real time.

One example of such a correlation is shown in Figure 4. The x-axis consists of the different molecular weight distributions. the numbers 1 to 3 are indices corresponding to M_w, M_n and M_z respectively. The z-axis shows the number value of each of the molecular weight distributions, while the y-axis shows the values of the input flow rates. These maps provide a means to chose set points for monomer inventories to achieve desired polymer properties. Similar plots were developed for the initiator and the other monomers.

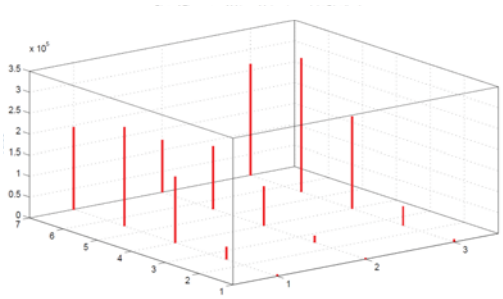


Fig. 4. M1-MWD correlation plot

Table 2. Simulation Results for varying M1

Flow rate of $M1$(kg/s)	M_w	M_n	M_z
1	4574	2439	6844
2.5	30003	15813	44928
4	158560	93652	219070
5.5	235980	149230	319110
7	195670	123340	264750

6. CONCLUSIONS

A kinetic model for free radical polymerization of the monomers in HPA/Sty/BA/BMA was developed and implemented in PREDICI. Kinetic parameters obtained from literature were found to predict the experimental data poorly. Further refinement was carried by implementing a parameter estimation algorithm based on nonlinear optimization. A sensitivity study and an estimability analysis were carried out to identify the set of estimable parameters. The estimable parameters were then optimized using a weighted, constrained non-linear least squares Gauss-Newton algorithm with backtracking line search. The optimized values of the parameters were found to yield better fits for the experimental data. However, there seems to be

[1] Due to space limitation we do not show the 4th monomer which is similar to the third and fourth.

[2] The simulated controller used a sampling time of 30 sec during the period when the setpoints were constant. It was decreased to one sample every 2 seconds for a brief period during the setpoint change

a trade-off between the two types of outputs (molecular weights and residual monomer concentrations) that may be optimized. It is possible to fit only either set of outputs very well, using this model. Certain features of the experimental data were not captured, indicating that the need for further refinement of the model by adding other reactions that are relevant to the process. An inventory control approach was proposed to force monomer concentrations to track some pre-determined set points. These set points can be appropriately selected based on the requirements using a correlation mapping such as the one performed in this study.

NOMENCLATURE

I $\equiv$ Initiator
M_i, $M_j \equiv$ Monomer
$S_k \equiv$ Solvent
$P_n^{i\cdot}$, $P_r^{i\cdot}$, $P_{n+r}^{i\cdot}$, $P_m^{i\cdot} \equiv$ Live Polymer Chain
D_n, D_r, D_m, $D_{n+r} \equiv$ Dead Polymer Chain
$\varphi_{ijk} \equiv$ Scaling factor

Sets

$\mathcal{M} \equiv \{M_1, M_2, M_3, M_4\} \equiv$ set of all monomers
$\mathcal{A} \equiv \{M_1, M_2, M_3\} \equiv$ set of Acrylate monomers
$\mathcal{MA} \equiv \{M_4\} \equiv$ the set of Methacrylate monomers
$\mathcal{P} \equiv \{P_n^{i\cdot}, P_r^{i\cdot}, P_m^{i\cdot}, P_{n+r}^{i\cdot}\}$ represents the set of Live Polymer Chains
$\mathcal{S} \equiv \{S_1, S_2\}$ represents the set of solvents
$\mathcal{D} \equiv \{D_n, D_r, D_m, D_{n+r}\}$ represents the set of dead polymer chains

Indices

$i, j \in \{1,2,3,4\}, \quad k \in \{1,2\}, \quad n, r, m \in [1, \ldots, \infty)$

REFERENCES

AkzoNobel (2006). *Initiators for High Polymers.* Akzo Nobel Polymer Chemicals, june edition.

Amrehn, H. (1977). Computer Control in Polymerization Industry. *Automatica*, 13, 533.

Asua, J.M. (2007). *Polymer Reaction Engineering.* Blackwell Publishing.

Beuermann, S. and Buback, M. (2002). Rate coefficients of free-radical polymerization deduced from pulsed laser experiments. *Prog. Polym. Sci*, 27, 191 – 254.

C. A. Farschman, K. P. Viswanath, B.E.Y. (1998). Process systems and inventory control. *AIChE Journal*, 44(8), 1841–1857.

Choi, T.J.C..K.Y. (1997). Discrete Optimal Control of Molecular Weight Distribution in a Batch Free Radical Polymerization Process. *Ind. Eng. Chem. Res.*, 36, 3676–3684.

Chow, C.D. (1975). Monomer Reactivity Ratio and Q-e Values for Copolymerization of Hydroxyalkyl Acrylates and 2-(1-Aziridinyl)ethyl Methacrylate with Styrene. *Journal of Polymer Science*, 13, 309 – 313.

Congling Quan, M.S. and Grady, M.C. (2003). Product Quality Improvement in a High-Temperature Free-Radical Polymerization Reactor. *Proceedings of the American Control Conference*, 3980 – 3985.

Curteanu, S. (2003). Modeling and Simulation of Free Radical Polymerization of Styrene under Semibatch Reactor conditions. *Central European Journal of Chemistry*, 1, 69 – 90.

Curteanu, S. and Bulacovschi, V. (2005). Free Radical Polymerization of Methyl Methacrylate: Modeling and Simulation under Semibatch and Nonisothermal Reactor Conditions. *Journal of Applied Polymer Science*, 74(11), 2561 – 2570.

D. Li, M.C.G. and Hutchinson, R.A. (2005). High-Temperature Semibatch Free Radical Copolymerization of Butyl Methacrylate and Butyl Acrylate. *Ind. Eng. Chem. Res.*, 44, 2506 – 2517.

Elicabe, G.E. and Meira, G.R. (1988). Estimation and Control in Polymerization Reactors - A Review. *Poly. Eng. & Sci.*, 28, 121.

H. Seki, M. Ogawab, S.O.K.A.M.O..W.Y. (2001). Industrial application of a nonlinear model predictive control to polymerization reactors. *Control Engineering Practice*, 9, 819 – 828.

Ham, G.E. (1964). *Copolymerization*, volume XVIII. Interscience Publishers.

K. Zhen Yao, B. M. Shaw, B.K.K.B.M..D.W.B. (2003). Modeling Ethylene/Butylene Copolymerization with Multi-Site Catalysis: Parameter Estimability and Experimental Design. *Polymer Reaction Engineering*, 11 (3), 563 – 588.

Khalil, H.K. (2002). *Nonlinear Systems, Third Ed.* Prentice Hall.

M. D. Díez, B. E. Ydstie, M.F.B.L. (2007). Inventory control of particulate processes. *Computers and Chemical Engineering*, 32, 46–67.

M. Ruszkowski, V. Garcia-Osorio, B.E.Y. (2005). Passivity based control of transport reaction systems. *AIChE Journal*, 51, 3147–3166.

MacGregor, J.F. (1986). Control of polymerization reactors. *Poly. Eng. & Sci.*, 31.

Maeder, S. and Gilbert, R.G. (1998). Measurement of transfer constant for butyl acrylate free-radical polymerization. *Macromolecules*, 31(14), 4410 – 4418.

Soroush, M. and Zambare, N. (2000). Nonlinear output feedback control of a class of polymerization reactors. *IEEE Trans. Control Systems Technology*, 8(2), 310–320.

W. R. Cluett, S.L.S..D.G.F. (1985). Adaptive Control of a Batch Reactor. *Chemical Engineering Communications*, 38, 67–78.

Ydstie, B.E. and Jiao, Y. (2006). Passivity based control of the float glass process: Multi-scale decomposition and real-time optimization of complex flows. *IEEE Control Systems Magazine*, 26(6), 64–72.

Simultaneous Regulation of Surface Roughness and Porosity in Thin Film Growth

Gangshi Hu [*] **Gerassimos Orkoulas** [*] **Panagiotis D. Christofides** [*,**,1]

[*] *Department of Chemical and Biomolecular Engineering, University of California, Los Angeles, CA 90095 USA.*
[**] *Department of Electrical Engineering, University of California, Los Angeles, CA 90095 USA.*

Abstract: This work focuses on simultaneous control of surface roughness and film porosity in a porous thin film deposition process modeled via kinetic Monte Carlo simulation on a triangular lattice. The microscopic model of the thin film growth process includes adsorption and migration processes. Vacancies and overhangs are allowed inside the film for the purpose of modeling thin film porosity. Appropriate closed-form dynamic models are first derived to describe the evolution of film surface roughness and porosity and used as the basis for the design of a model predictive control algorithm that includes penalty on the deviation of surface roughness and film porosity from their respective set-point values. Closed-loop simulations demonstrate that when simultaneous control of surface roughness and porosity is carried out, a balanced trade-off is obtained in the closed-loop system between the two control objectives of surface roughness and porosity regulation.

Keywords: thin film processing, roughness, porosity, model predictive control

1. INTRODUCTION

Thin film deposition processes play an important role in the semiconductor industry. Thin film microstructure, including surface roughness and film porosity, strongly affects the electrical and mechanical properties of thin films and of the resulting devices. Motivated by this, recent research efforts on modeling and control of thin film microstructure have focused mostly on thin film surface roughness on the basis of microscopic thin film growth models which utilize a square lattice. Specifically, kinetic Monte Carlo (kMC) models based on a square lattice and utilizing the solid-on-solid (SOS) approximation for deposition were initially employed to develop an effective methodology to describe the evolution of film microstructure and design feedback control laws for thin film surface roughness (Lou and Christofides (2003); Christofides et al. (2008)). This control methodology was successfully applied to surface roughness control of: a) a gallium arsenide (GaAs) deposition process (Lou and Christofides (2004)), and b) a multi-species deposition process with long range interactions (Ni and Christofides (2005a)). Furthermore, a method that couples partial differential equation (PDE) models and kMC models was developed for computationally efficient multiscale optimization of thin film growth (Varshney and Armaou (2005)). However, kMC models are not available in closed-form and this limitation restricts the use of kMC models for system-level analysis and design of model-based feedback control systems. To overcome this problem, model identification of linear deterministic models from outputs of kMC simulators was used for controller design using linear control theory (Siettos et al. (2003); Armaou et al. (2004)). However, deterministic models are only effective in controlling the expected values of macroscopic variables, i.e., the first-order statistical moments of the microscopic distribution. For higher statistical moments of the microscopic distributions such as the surface roughness (the second moment of height distribution on a lattice), deterministic models are not sufficient, and stochastic differential equation (SDE) models may be needed.

SDEs arise naturally in the modeling of surface morphology of ultra thin films in a variety of thin film preparation processes (Edwards and Wilkinson (1982); Villain (1991); Vvedensky et al. (1993)). Advanced control methods based on SDEs have been developed to address the need of model-based feedback control of thin film microstructure. Specifically, methods for state feedback control of surface roughness based on linear (Lou and Christofides (2005); Ni and Christofides (2005b)) and nonlinear (Lou and Christofides (2008)) SDE models have been developed. However, state feedback control assumes full knowledge of the surface morphology at all times, which may be a restrictive requirement in certain practical applications. To this end, output feedback control of surface roughness was recently developed (Hu et al. (2008)) by incorporating a Kalman-Bucy type filter, which utilizes information from a finite number of noisy measurements.

In the context of modeling of thin film porosity, kMC models have been widely used to model the evolution of porous thin films in many deposition processes and to investigate the influence of the macroscopic parameters on the porous thin film microstructure (Wang and Clancy (1998); Zhang et al. (2004)). Deterministic and stochastic ordinary differential equation (ODE) models of film porosity were recently developed (Hu et al. (2009a)) to model the evolution of film porosity and its fluctuation and design model predictive control (MPC) algorithms to control film porosity to a desired level and reduce run-to-run porosity variability. Despite recent significant efforts on modeling and control of surface roughness and film porosity, simultaneous regulation of surface roughness and film porosity within a unified control framework has not been investigated.

[1] Corresponding author: Tel: +1(310)794-1015; Fax: +1(310)206-4107; (e-mail: pdc@seas.ucla.edu). Financial support from NSF, CBET-0652131, is gratefully acknowledged.

Motivated by these considerations, the present work focuses on simultaneous regulation of surface roughness and film porosity in a porous thin film deposition process modeled via kMC simulation on a triangular lattice. The definition of surface height profile is first introduced and the dynamics of surface height of the thin film are described by an Edwards-Wilkinson (EW)-type equation. Subsequently, an appropriate definition of film site occupancy ratio (SOR) is introduced to represent the porosity and a deterministic ODE model is derived to describe the time evolution of film SOR. The model parameters are estimated on the basis of data obtained from the kMC simulator of the deposition process using least-square methods. The developed dynamic models are used as the basis for the design of a model predictive control algorithm that includes penalty on the deviation of surface roughness square and film SOR from their respective set-point values. Simulation results demonstrate the applicability and effectiveness of the proposed modeling and control approach in the context of the deposition process under consideration.

2. PROCESS DESCRIPTION AND MODELING

2.1 On-lattice kinetic Monte Carlo model of film growth

The thin film growth process considered in this work includes two microscopic processes: an adsorption process, in which particles are incorporated into the film from the gas phase, and a migration process, in which surface particles move to adjacent sites (Wang and Clancy (1998); Levine and Clancy (2000); Yang et al. (1997)). Specifically, the film growth model used in this work is an on-lattice kMC model in which all particles occupy discrete lattice sites. The on-lattice kMC model is valid for temperatures $T < 0.5T_m$, where T_m is the melting point of the crystal. At high temperatures ($T \lesssim T_m$), the particles cannot be assumed to be constrained on the lattice sites and the on-lattice model is not valid. In this work, a triangular lattice is selected to represent the crystalline structure of the film, as shown in Fig.1. All particles are modeled as identical hard disks and the centers of the particles deposited on the film are located on the lattice sites. The diameter of the particles equals the distance between two neighboring sites. The width of the lattice is fixed so that the lattice contains a fixed number of sites in the lateral direction. The new particles are always deposited vertically from the top side of the lattice where the gas phase is located; see Fig.1. Particle deposition results in film growth in the direction normal to the lateral direction. The direction normal to the lateral direction is thus designated as the growth direction. The number of sites in the lateral direction is defined as the lattice size and is denoted by L. The lattice parameter, a, which is defined as the distance between two neighboring sites and equals the diameter of a particle (all particles have the same diameter), determines the lateral extent of the lattice, La.

The number of nearest neighbors of a site ranges from zero to six, the coordination number of the triangular lattice. A site with no nearest neighbors indicates an unadsorbed particle in the gas phase (i.e., a particle which has not been deposited on the film yet). A particle with six nearest neighbors is associated with an interior particle that is fully surrounded by other particles and cannot migrate. A particle with two to five nearest neighbors is possible to diffuse to an unoccupied neighboring site with a probability that depends on its local environment. In the triangular lattice, a particle with only one nearest neighbor is considered unstable and is subject to instantaneous surface relaxation.

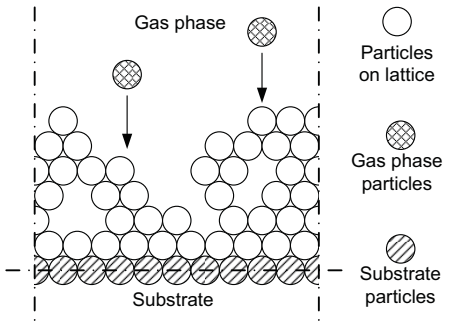


Fig. 1. Thin film growth process on a triangular lattice.

In the simulation, a bottom layer in the lattice is initially set to be fully packed and fixed, as shown in Fig.1. There are no vacancies in this layer and the particles in this layer cannot migrate. This layer acts as the substrate for the deposition and is not counted in the computation of the number of the deposited particles, i.e., this fixed layer does not influence the film surface roughness and porosity (see Section 2.2 below). All microscopic processes (Monte Carlo events) are assumed to be Poisson processes. These Monte Carlo events occur randomly with probabilities proportional to their respective rates. The events are executed instantaneously upon selection and the state of the lattice remains unchanged between two consecutive events. The specific rules used to carry out the adsorption and migration processes and their simulation are discussed in detail in Hu et al. (2009b) and are not presented here due to space limitations.

2.2 Definitions of surface roughness and site occupancy ratio

Utilizing the continuous-time Monte Carlo algorithm, simulations of the kMC model of a porous silicon thin film growth process can be carried out. Snapshots of film microstructure, i.e., the configurations of particles within the triangular lattice, are obtained from the kMC model at various time instants during process evolution. To quantitatively evaluate the thin film microstructure, two variables, surface roughness and film porosity, are introduced in this subsection.

Surface roughness, which measures the texture of thin film surface, is represented by the root mean square (RMS) of the surface height profile of the thin film. Determination of surface height profile is slightly different in the triangular lattice model compared to a SOS model. In the SOS model, the surface of thin film is naturally described by the positions of the top particles of each column. In the triangular lattice model, however, due to the existence of vacancies and overhangs, the definition of film surface needs further clarification. Specifically, taking into account practical considerations of surface roughness measurements, the surface height profile of a triangular lattice model is defined based on the particles that can be reached from abobe in the vertical direction, as shown in Fig.2. In this definition, a particle is considered as a surface particle only if it is not blocked by the particles in both neighboring columns. Therefore, the surface height profile of a porous thin film is the line that connects the sites that are occupied by the surface particles. With this definition, the surface height profile can be treated as a function of the spatial coordinate. Surface roughness, as a measurement of the surface texture, is defined as the standard deviation of the surface height profile from its average height. The definition expression of surface roughness is given later in Section 3.1.

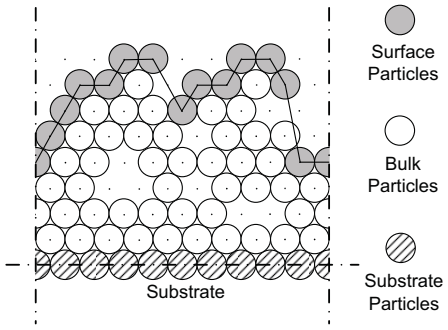


Fig. 2. Definition of surface height profile. A surface particle is a particle that is not blocked by particles from both of its neighboring columns in the vertical direction.

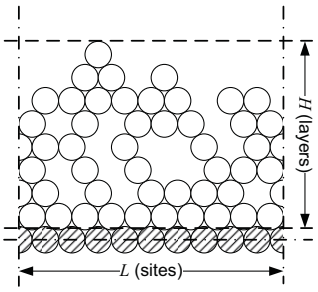


Fig. 3. Illustration of the definition of film SOR of Eq.1.

In addition to film surface roughness, the film site occupancy ratio (SOR) is introduced to represent the extent of the porosity inside the thin film. The mathematical expression of film SOR is defined as follows:

$$\rho = \frac{N}{LH} \tag{1}$$

where ρ denotes the film SOR, N is the total number of deposited particles on the lattice, L is the lattice size, and H denotes the number of deposited layers. Note that the deposited layers are the layers that contain only deposited particles and do not include the initial substrate layers. The variables in the definition expression of Eq.1 can be found in Fig.3. Since each layer contains L sites, the total number of sites in the film that can be contained within the H layers is LH. Thus, film SOR is the ratio of the occupied lattice sites, N, over the total number of available sites, LH. Film SOR ranges from 0 to 1. Specifically, $\rho = 1$ denotes a fully occupied film with a flat surface. The value of zero is assigned to ρ at the beginning of the deposition process since there are no particles deposited on the lattice.

3. DYNAMIC MODEL CONSTRUCTION AND PARAMETER ESTIMATION

3.1 Edwards-Wilkinson-type equation of surface height

An Edwards-Wilkinson (EW)-type equation can be used to describe the surface height evolution in many microscopic processes that involve thermal balance between adsorption (deposition) and migration (diffusion). In this work, an EW-type equation is chosen to describe the dynamics of the fluctuation of surface :

$$\frac{\partial h}{\partial t} = r_h + \nu \frac{\partial^2 h}{\partial x^2} + \xi(x,t) \tag{2}$$

subject to PBCs:

$$h(-\pi,t) = h(\pi,t), \quad \frac{\partial h}{\partial x}(-\pi,t) = \frac{\partial h}{\partial x}(\pi,t) \tag{3}$$

and the initial condition:

$$h(x,0) = h_0(x) \tag{4}$$

where $x \in [-\pi,\pi]$ is the spatial coordinate, t is the time, r_h and ν are the model parameters, and $\xi(x,t)$ is a Gaussian white noise with the following expressions for its mean and covariance:

$$\begin{aligned} \langle \xi(x,t) \rangle &= 0 \\ \left\langle \xi(x,t)\xi(x',t') \right\rangle &= \sigma^2 \delta(x-x')\delta(t-t') \end{aligned} \tag{5}$$

where σ^2 is a parameter which measures the intensity of the Gaussian white noise and $\delta(\cdot)$ denotes the standard Dirac delta function.

To proceed with model parameter estimation and control design, a stochastic ODE approximation of Eq.2 is first derived using Galerkin's method. Consider the eigenvalue problem of the linear operator of Eq.2, which takes the form:

$$\begin{aligned} A\bar{\phi}_n(x) &= \nu \frac{d^2\bar{\phi}_n(x)}{dx^2} = \lambda_n \bar{\phi}_n(x) \\ \bar{\phi}_n(-\pi) &= \bar{\phi}_n(\pi), \quad \frac{d\bar{\phi}_n}{dx}(-\pi) = \frac{d\bar{\phi}_n}{dx}(\pi) \end{aligned} \tag{6}$$

where λ_n denotes an eigenvalue and $\bar{\phi}_n$ denotes an eigenfunction. A direct computation of the solution of the above eigenvalue problem yields $\lambda_0 = 0$ with $\psi_0 = 1/\sqrt{2\pi}$, and $\lambda_n = -\nu n^2$ (λ_n is an eigenvalue of multiplicity two) with eigenfunctions $\phi_n = (1/\sqrt{\pi})\sin(nx)$ and $\psi_n = (1/\sqrt{\pi})\cos(nx)$ for $n = 1,\cdots,\infty$. Note that the $\bar{\phi}_n$ in Eq.6 denotes either ϕ_n or ψ_n. For fixed positive value of ν, all eigenvalues (except the zero-th eigenvalue) are negative and the distance between two consecutive eigenvalues (i.e. λ_n and λ_{n+1}) increases as n increases.

To this end, the solution of Eq.2 is expanded in an infinite series in terms of the eigenfunctions as follows:

$$h(x,t) = \sum_{n=1}^{\infty} \alpha_n(t)\phi_n(x) + \sum_{n=0}^{\infty} \beta_n(t)\psi_n(x) \tag{7}$$

where $\alpha_n(t)$, $\beta_n(t)$ are time-varying coefficients. Substituting the above expansion for the solution, $h(x,t)$, into Eq.2 and taking the inner product with the adjoint eigenfunctions, $\phi_n^*(x) = (1/\sqrt{\pi})\sin(nx)$ and $\psi_n^*(x) = (1/\sqrt{\pi})\cos(nx)$, the following system of infinite stochastic ODEs is obtained:

$$\begin{aligned} \frac{d\beta_0}{dt} &= \sqrt{2\pi} r_h + \xi_\beta^0(t) \\ \frac{d\alpha_n}{dt} = \lambda_n \alpha_n + \xi_\alpha^n(t), \frac{d\beta_n}{dt} &= \lambda_n \beta_n + \xi_\beta^n(t), n = 1,\ldots,\infty \end{aligned} \tag{8}$$

where

$$\xi_\alpha^n(t) = \int_{-\pi}^{\pi} \xi(x,t)\phi_n^*(x)dx, \quad \xi_\beta^n(t) = \int_{-\pi}^{\pi} \xi(x,t)\psi_n^*(x)dx. \tag{9}$$

The covariances of $\xi_\alpha^n(t)$ and $\xi_\beta^n(t)$ can be computed as follows:

$\langle \xi_\alpha^n(t)\xi_\alpha^n(t') \rangle = \sigma^2\delta(t-t')$ and $\left\langle \xi_\beta^n(t)\xi_\beta^n(t') \right\rangle = \sigma^2\delta(t-t')$.

Since the stochastic ODE system is linear, the analytical solution of state variance can be obtained from a direct computation as follows:

$$\begin{aligned} \left\langle \alpha_n^2(t) \right\rangle &= \frac{\sigma^2}{2\nu n^2} + \left(\left\langle \alpha_n^2(t_0) \right\rangle - \frac{\sigma^2}{2\nu n^2} \right) e^{-2\nu n^2 (t-t_0)} \\ \left\langle \beta_n^2(t) \right\rangle &= \frac{\sigma^2}{2\nu n^2} + \left(\left\langle \beta_n^2(t_0) \right\rangle - \frac{\sigma^2}{2\nu n^2} \right) e^{-2\nu n^2 (t-t_0)} \\ & n = 1,2,\ldots,\infty \end{aligned} \tag{10}$$

where $\langle \alpha_n^2(t_0) \rangle$ and $\langle \beta_n^2(t_0) \rangle$ are the state variances at time t_0. The analytical solution of state variance of Eq.10 will be used in the parameter estimation and the MPC design.

When the dynamic model of surface height profile is determined, surface roughness of the thin film is defined as the standard deviation of the surface height profile from its average height and is computed as follows:

$$r(t) = \sqrt{\frac{1}{2\pi}\int_{-\pi}^{\pi} [h(x,t) - \bar{h}(t)]^2 dx} \tag{11}$$

where $\bar{h}(t) = \frac{1}{2\pi}\int_{-\pi}^{\pi} h(x,t)dx$ is the averaged surface height. According to Eq.7, we have $\bar{h}(t) = \beta_0(t)\psi_0$. Therefore, $\langle r^2(t) \rangle$ can be rewritten in terms of $\langle \alpha_n^2(t) \rangle$ and $\langle \beta_n^2(t) \rangle$ as follows:

$$\langle r^2(t) \rangle = \frac{1}{2\pi}\left\langle \int_{-\pi}^{\pi} (h(x,t) - \bar{h}(t))^2 dx \right\rangle$$

$$= \frac{1}{2\pi}\left\langle \sum_{i=1}^{\infty} (\alpha_i^2(t) + \beta_i^2(t)) \right\rangle = \frac{1}{2\pi}\sum_{i=1}^{\infty} \left[\langle \alpha_i^2(t) \rangle + \langle \beta_i^2(t) \rangle \right] \tag{12}$$

where $\bar{h} = \frac{1}{2\pi}\int_{-\pi}^{\pi} h(x,t)dx = \beta_0(t)\psi_0$ is the average of surface height. Thus, Eq.12 provides a direct link between the state variance of the infinite stochastic ODEs of Eq.8 and the expected surface roughness of the thin film. Note that the model parameter r_h does not appear in the expression of surface roughness, since only the zeroth state, β_0, is affected by r_h but this state is not included in the computation of the expected surface roughness square of Eq.12.

3.2 Deterministic dynamic model of film site occupancy ratio

Since film porosity is another control objective, a dynamic model is necessary in the MPC formulation to describe the evolution of film porosity, which is represented by the film SOR of Eq.1. The dynamics of the expected value of the film SOR evolution are approximately described by a linear first-order deterministic ODE as follows:

$$\tau \frac{d\langle \rho(t) \rangle}{dt} = \rho^{ss} - \langle \rho(t) \rangle \tag{13}$$

where t is the time, τ is the time constant and ρ^{ss} is the steady-state value of the film SOR. The deterministic ODE system of Eq.13 is subject to the following initial condition:

$$\langle \rho(t_0) \rangle = \rho_0 \tag{14}$$

where t_0 is the initial time and ρ_0 is the initial value of the film SOR. Note that ρ_0 is a deterministic variable, since ρ_0 refers to the film SOR at $t = t_0$. From Eqs.13 and 14, it follows that

$$\langle \rho(t) \rangle = \rho^{ss} + (\rho_0 - \rho^{ss})\, e^{-(t-t_0)/\tau}. \tag{15}$$

3.3 Parameter estimation

Referring to the EW equation of Eq.2 and the deterministic ODE model of Eq.13, there are several model parameters, ν, σ^2, ρ^{ss} and τ, that need to be determined as functions of the substrate temperature. These parameters describe the dynamics of surface height and of film SOR and can be estimated by comparing the predicted evolution profiles from the dynamic models of Eqs.2 and 13 and the ones from the kMC simulation of the deposition process in a least-square sense (Hu et al. (2009a,b)).

Since surface roughness is a control objective, we choose the expected surface roughness square of Eq.12 as the output for the parameter estimation of the EW equation of Eq.2. Thus, the model coefficients, ν and σ^2, can be obtained by solving the minimization problem as follows:

$$\min_{\nu,\sigma^2} \sum_{i=1}^{n_1} \left[\langle r^2(t) \rangle - \frac{1}{2\pi}\sum_{i=1}^{\infty} \left(\langle \alpha_i^2(t) \rangle + \langle \beta_i^2(t) \rangle \right) \right]^2 \tag{16}$$

where n_1 is the number of the data samplings of surface height profile and surface roughness from the kMC simulations. The predictions of model state variance, $\langle \alpha_i^2(t) \rangle$ and $\langle \beta_i^2(t) \rangle$, can be solved from the analytical solution of Eq.10.

With respect to the parameters of the equation for film porosity, ρ^{ss} and τ can be estimated similarly from the solutions of Eq.15 as follows:

$$\min_{\rho^{ss},\tau} \sum_{i=1}^{n_2} \left[\langle \rho(t_i) \rangle - \left(\rho^{ss} + (\rho_0 - \rho^{ss})\, e^{-(t-t_0)/\tau} \right) \right]^2 \tag{17}$$

where n_2 is the number of the data samplings of film SOR from the kMC simulations. We note that since the dynamic models of film surface height and film SOR may have different dynamics, different numbers of data samplings at different time instants may be used to estimate the parameters of the dynamic models.

The data used for the parameter estimation are obtained from the open-loop kMC simulation of the thin film growth process. The process parameters, i.e., the substrate temperature and the adsorption rate, are fixed during each open-loop simulation. The predictions from the dynamic models with the estimated parameters are close to the open-loop simulation profiles. Detailed data and plots can be found in Hu et al. (2009b).

The parameters that are estimated from fixed operating conditions are suitable for the feedback control design in this work. This is because the control input in the MPC formulation is piecewise, i.e., the manipulated substrate temperature remains constant between two consecutive sampling times, and thus, the dynamics of the microscopic process can be predicted from the dynamic models with estimated parameters. The dependence of the model parameters on substrate temperature is used in the formulation of the model predictive controller in the next section. Thus, parameter estimation from open-loop kMC simulation results of the thin film growth process for a variety of operation conditions is performed to obtain the dependence of the model coefficients on substrate temperature. In this work, the deposition rate for all simulations is fixed at 1 layer/s. The range of T is between 300 K and 800 K, which is from room temperature to the upper limit of the allowable temperature for a valid on-lattice kMC model of silicon film. The dependence of the model parameters on the substrate temperature can be found in Hu et al. (2009b).

4. MODEL PREDICTIVE CONTROL DESIGN

We consider the problem of regulation of surface roughness and of film SOR to desired levels within a model predictive control framework. State feedback control is considered in this work, i.e., the surface height profile and the value of film SOR are assumed to be available to the controller. Real-time film roughness and SOR can be estimated from in-situ thin film thickness measurements (Buzea and Robbie, 2005) in combination with off-line film porosity measurements. Since surface roughness and film SOR are stochastic variables, the expected values, $\langle r(t)^2 \rangle$ and $\langle \rho \rangle$, are chosen as the control objectives. The substrate temperature is used as the manipulated input and the deposition rate is fixed at a certain value, W_0, during the entire closed-loop simulation. To account for a number of prac-

tical considerations, several constraints are added to the control problem. First, there is a constraint on the range of variation of the substrate temperature. This constraint ensures validity of the on-lattice kMC model. Another constraint is imposed on the rate of change of the substrate temperature to account for actuator limitations. The control action at a time t is obtained by solving a finite-horizon optimal control problem. The cost function in the optimal control problem includes penalty on the deviation of $\langle r^2 \rangle$ and $\langle \rho \rangle$ from their respective set-point values. Different weighting factors are assigned to the penalties of the surface roughness and of the film SOR. Surface roughness and film SOR have very different magnitudes, ($\langle r^2 \rangle$ ranges from 1 to 10^2 and $\langle \rho \rangle$ ranges from 0 to 1). Therefore, relative deviations are used in the formulation of the cost function to make the magnitude of the two terms comparable. The optimization problem is subject to the dynamics of the surface height of Eq.2 and of the film SOR of Eq.13. The optimal temperature profile is calculated by solving a finite-dimensional optimization problem in a receding horizon fashion. Specifically, the MPC problem is formulated as follows:

$$\begin{gathered} \min_{T_1,\ldots,T_i,\ldots,T_p} J = \sum_{i=1}^{p} \Big\{ q_{r^2,i} \left[(r^2_{set} - \langle r^2(t_i) \rangle)/r^2_{set} \right]^2 \\ + q_{\rho,i} \left[(\rho_{set} - \langle \rho(t_i) \rangle)/\rho_{set} \right]^2 \Big\} \\ \text{subject to} \\ \frac{\partial h}{\partial t} = r_h + \nu \frac{\partial^2 h}{\partial x^2} + \xi(x,t), \tau \frac{d\langle \rho(t) \rangle}{dt} = \rho^{ss} - \langle \rho(t) \rangle \\ T_{min} < T_i < T_{max}, |(T_{i+1} - T_i)/\Delta| \le L_T \\ i = 1, 2, \ldots, p \end{gathered} \tag{18}$$

where t is the current time, Δ is the sampling time, p is the number of prediction steps, $p\Delta$ is the specified prediction horizon, t_i, $i = 1, 2, \ldots, p$, is the time of the ith prediction step ($t_i = t + i\Delta$), respectively, T_i, $i = 1, 2, \ldots, p$, is the substrate temperature at the ith step ($T_i = T(t + i\Delta)$), respectively, W_0 is the fixed deposition rate, $q_{r^2,i}$ and $q_{\rho,i}$, $i = 1, 2, \ldots, p$, are the weighting penalty factors for the deviations of $\langle r^2 \rangle$ and $\langle \rho \rangle$ from their respective set-points at the ith prediction step, T_{min} and T_{max} are the lower and upper bounds on the substrate temperature, respectively, and L_T is the limit on the rate of change of the substrate temperature.

The optimal set of control actions, (T_1, T_2, …, T_p), is obtained from the solution of the multi-variable optimization problem of Eq.18, and only the first value of the manipulated input trajectory, T_1, is applied to the deposition process during the time interval $(t, t+\Delta)$. At time $t+\Delta$, a new measurement of ρ and h is received and the MPC problem of Eq.18 is solved for the next control input trajectory.

The MPC formulation proposed in Eq.18 is developed on the basis of the EW equation of surface height and the deterministic ODE model of the film SOR. The EW equation, which is a distributed parameter dynamic model, contains infinite dimensional stochastic states. Therefore, it leads to a model predictive controller of infinite order that cannot be realized in practice (i.e., the practical implementation of such a control algorithm will require the computation of infinite sums which cannot be done by a computer). To this end, a finite dimensional approximation of the EW equation of order $2m$, derived using modal decomposition, is used in the simulations below.

5. SIMULATION RESULTS

In this section, the model predictive controller is applied to the kMC model of the thin film growth process described in Section 2. The value of the substrate temperature is obtained from the solution of the MPC problem at each sampling time and is applied to the closed-loop system until the next sampling time. The optimization problem is solved using a local constrained minimization algorithm using a broad set of initial guesses.

The constraint on the rate of change of the substrate temperature is imposed onto the optimization problem, which is realized in the optimization process in the following way:

$$\left| \frac{T_{i+1} - T_i}{\Delta} \right| \le L_T \Rightarrow T_i - L_T\Delta \le T_{i+1} \le T_i + L_T\Delta \qquad i = 1, 2, \ldots, p. \tag{19}$$

The desired values (set-point values) in the closed-loop simulations are $r^2_{set} = 10.0$ and $\rho_{set} = 0.95$. The order of finite-dimensional approximation of the EW equation in the MPC formulation is $m = 20$. The deposition rate is fixed at 1 layer/s and initial temperature of 600 K. The variation of temperature is from 400 K to 700 K. The maximum of change of the temperature is $L_T = 10$ K/s. The sampling time is fixed at $\Delta = 1$ s. The number of prediction steps is set to be $p = 5$. The simulation duration is determined on the basis of a desired film thickness and the fixed adsorption rate and is chosen as 1000 s for the closed-loop simulations in this work. All expected values are obtained from 1000 independent simulation runs.

Closed-loop simulations of separately regulating film surface roughness and porosity are first carried out. In these control problems, the control objective is to only regulate one of the control variables, i.e., either surface roughness or film SOR, to a desired level. The cost functions of these problems contain only penalty on the error either of the expected surface roughness square, or of the expected film SOR, from their set-point values. The corresponding MPC formulations can be realized by assigning different values to the penalty weighting factors, $q_{r^2,i}$ and $q_{\rho,i}$.

In the roughness-only control problem, the weighting factors take the following values: $q_{r^2,i} = 1$ and $q_{\rho,i} = 0$, $i = 1, 2, \ldots, p$. Fig.4 shows the closed-loop simulation results of the roughness-only control problem. From Fig.4, we can see that the expected surface roughness square is successfully regulated at the desire level, 10. Since no penalty is included on the error of the expected film SOR, the final value of expected film SOR at the end of the simulation, $t = 1000$ s, is 0.988, which is far from the desired film SOR, 0.95.

In the SOR-only control problem, the weighting factors are assigned as: $q_{r^2,i} = 0$ and $q_{\rho,i} = 1$, $i = 1, 2, \ldots, p$. Fig.5 shows the closed-loop simulation results of the SOR-only control problem. Similar to the results of the roughness-only control problem, the desired value of expected film SOR, 0.95, is approached at large times. However, since the error from the expected surface roughness square is not considered in the cost function, $\langle r^2 \rangle$ reaches a very high level around 125 at the end of the simulation.

Finally, closed-loop simulations of simultaneous regulation of surface roughness and film SOR are carried out by assigning non-zero values to both penalty weighting factors. Specifically, $q_{r^2,1} = q_{r^2,2} = \cdots = q_{r^2,p} = 1$ and $q_{\rho,1} = q_{\rho,2} = \cdots = q_{\rho,p} = q_{SOR}$ and q_{SOR} varies from 1 to 10^4. Since substrate temperature is the only manipulated input, the desired-values of r^2_{set} and ρ_{set} cannot be achieved simultaneously. With different assignments of penalty weighting factors, the MPC evaluates and strikes a balance between the two set-points. Fig.6 shows the expected

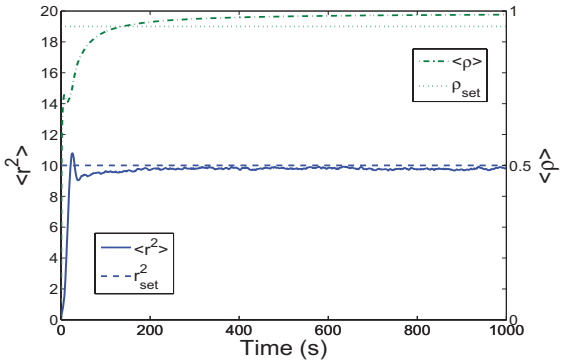


Fig. 4. Profiles of the expected values of surface roughness square (solid line) and of the film SOR (dash-dotted line) under closed-loop operations with cost function including only penalty on surface roughness.

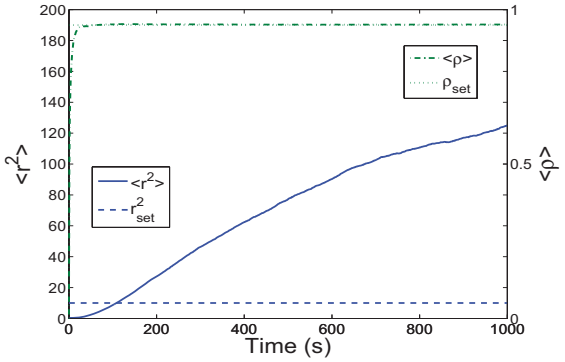


Fig. 5. Profiles of the expected values of surface roughness square (solid line) and of the film SOR (dash-dotted line) under closed-loop operation with cost function including only penalty on the film SOR.

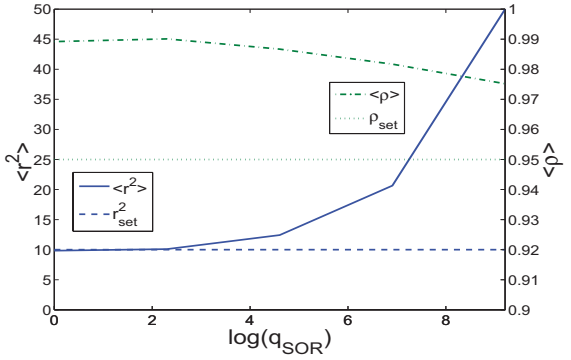


Fig. 6. Profiles of the expected values of surface roughness square (solid line) and of the film SOR (dash-dotted line) at the end of the closed-loop simulations ($t = 1000$ s) with the following penalty weighting factors: $q_{r^2,i}$ fixed at 1 for all i and for different values of q_{SOR}.

values of r^2_{set} and ρ_{set} at the end of closed-loop simulations of the simultaneous control problem with respect to different weighting factors. It is clear from Fig.6 that as the weighting on expected film SOR increases, the expected film SOR approaches its set-point value of 0.95, while the expected surface roughness square deviates from its set-point value of 10.

REFERENCES

Armaou, A., Siettos, C.I., and Kevrekidis, I.G. (2004). Time-steppers and 'coarse' control of distributed microscopic processes. *International Journal of Robust and Nonlinear Control*, 14, 89–111.

Buzea, C. and Robbie, K. (2005). State of the art in thin film thickness and deposition rate monitoring sensors. *Reports on Progress in Physics*, 68, 385–409.

Christofides, P.D., Armaou, A., Lou, Y., and Varshney, A. (2008). *Control and Optimization of Multiscale Process Systems*. Birkhäuser, Boston.

Edwards, S.F. and Wilkinson, D.R. (1982). The surface statistics of a granular aggregate. *Proceedings of the Royal Society of London Series A - Mathematical Physical and Engineering Sciences*, 381, 17–31.

Hu, G., Lou, Y., and Christofides, P.D. (2008). Dynamic output feedback covariance control of stochastic dissipative partial differential equations. *Chemical Engineering Science*, 63, 4531–4542.

Hu, G., Orkoulas, G., and Christofides, P.D. (2009a). Modeling and control of film porosity in thin film deposition. *Chemical Engineering Science,* accepted.

Hu, G., Orkoulas, G., and Christofides, P.D. (2009b). Simultaneous regulation of surface roughness and porosity in thin film deposition. *Industrial & Engineering Chemistry Research,* accepted.

Levine, S.W. and Clancy, P. (2000). A simple model for the growth of polycrystalline Si using the kinetic Monte Carlo simulation. *Modelling and Simulation in Materials Science and Engineering*, 8, 751–762.

Lou, Y. and Christofides, P.D. (2003). Estimation and control of surface roughness in thin film growth using kinetic Monte-Carlo models. *Chemical Engineering Science*, 58, 3115–3129.

Lou, Y. and Christofides, P.D. (2004). Feedback control of surface roughness of GaAs (001) thin films using kinetic Monte Carlo models. *Computers & Chemical Engineering*, 29, 225–241.

Lou, Y. and Christofides, P.D. (2005). Feedback control of surface roughness using stochastic PDEs. *AIChE Journal*, 51, 345–352.

Lou, Y. and Christofides, P.D. (2008). Nonlinear feedback control of surface roughness using a stochastic PDE: Design and application to a sputtering process. *Industrial & Engineering Chemistry Research*, 45, 7177–7189.

Ni, D. and Christofides, P.D. (2005a). Dynamics and control of thin film surface microstructure in a complex deposition process. *Chemical Engineering Science*, 60, 1603–1617.

Ni, D. and Christofides, P.D. (2005b). Multivariable predictive control of thin film deposition using a stochastic PDE model. *Industrial & Engineering Chemistry Research*, 44, 2416–2427.

Siettos, C.I., Armaou, A., Makeev, A.G., and Kevrekidis, I.G. (2003). Microscopic/stochastic timesteppers and "coarse" control: a kMC example. *AIChE Journal*, 49, 1922–1926.

Varshney, A. and Armaou, A. (2005). Multiscale optimization using hybrid PDE/kMC process systems with application to thin film growth. *Chemical Engineering Science*, 60, 6780–6794.

Villain, J. (1991). Continuum models of crystal growth from atomic beams with and without desorption. *Journal de Physique I*, 1, 19–42.

Vvedensky, D.D., Zangwill, A., Luse, C.N., and Wilby, M.R. (1993). Stochastic equations of motion for epitaxial growth. *Physical Review E*, 48, 852–862.

Wang, L. and Clancy, P. (1998). A kinetic Monte Carlo study of the growth of Si on Si(100) at varying angles of incident deposition. *Surface Science*, 401, 112–123.

Yang, Y.G., Johnson, R.A., and Wadley, H.N. (1997). A Monte Carlo simulation of the physical vapor deposition of nickel. *Acta Materialia*, 45, 1455–1468.

Zhang, P., Zheng, X., Wu, S., Liu, J., and He, D. (2004). Kinetic Monte Carlo simulation of Cu thin film growth. *Vacuum*, 72, 405–410.

A Strategy for Controlling Acetaldehyde Content in an Industrial Plant of Bioethanol

Fabio R. M. Batista*. Antonio J. A. Meirelles**

** Laboratory EXTRAE, Department of Food Engineering, Faculty of Food Engineering, University of Campinas - UNICAMP, Brazil (e-mail: f.fabio.batista@gmail.com).*

*** Laboratory EXTRAE, Department of Food Engineering, Faculty of Food Engineering, University of Campinas - UNICAMP, Brazil (Phone: 55-19- 3521-4037, e-mail: tomze@fea.unicamp.br).*

Abstract: This work presents a strategy for controlling acetaldehyde content in Brazilian bioethanol, based in simulation results of a typical industrial distillation plant. The major problem of acetaldehyde in bioethanol is that, during the storage period, it can oxidize to acetic acid, increasing fuel acidity above the legislation limit. This work tested, by dynamic simulation, simple loops to control acetaldehyde in bioethanol. The dynamic simulation generated a disturbance in the wine to be distilled by increasing acetaldehyde content, and verified how those loops were able to control the acetaldehyde level in bioethanol. Two different column system configurations were investigated. The first one includes a degassing system and a second one that produces pasteurized alcohol without or with a degassing system. Suggestions for the best control system of acetaldehyde contamination in bioethanol were formulated according to the acetaldehyde level in the wine.

Keywords: Fuel ethanol, bioethanol, dynamic simulation, degassing system, aspen plus.

1. INTRODUCTION

There is an increasing interest in bioethanol as a renewable energy source as well as a commodity to be used in other industrial branches, such as the chemical, pharmaceutical, and beverage industries. Brazil is one of the largest bioethanol producers and the largest exporter. For more than 30 years bioethanol is used directly as a biofuel, in this case with a concentration close to the azeotropic one, or added to petrol and, in this last case, it should be anhydrous. The rapid increase in its use as biofuel, the increase of its exports and of its use in other industrial branches is requiring a better control of product quality. Several minor components are generated during bioethanol production by fermentation and most of them are contaminants present in the end product. Although ethanol distillation is a largely investigated subject, most of the research works focus on energy consumption, alternative dehydration techniques and control strategies for separating the binary mixture ethanol-water, not taking into account the series of minor components that influence the distillation process. Those research works also rarely consider the peculiarities of the column systems used for ethanol distillation in the industrial practice.

Some recent works are applying simulations tools in order to investigate spirits and bioethanol distillation, taking into account at least part of the complexity of the multicomponent alcoholic mixture and of the industrial equipments used for its distillation. GAISER et al. (2002) used the commercial software Aspen Plus for simulating a continuous industrial unit for whiskey distillation, validating the results against industrial data. MEIRELLES et al. (2008) simulated a continuous distillation column for spirits production from sugar cane fermented must. DECLOUX and COUSTEL (2005) simulated a typical distillation plant for neutral alcohol production, using the software ProSim Plus. Neutral alcohol is a very pure ethanol product that requires a series of distillation columns to be produced.

Taking into account the increasing importance of bioethanol and the largely untreated subject of controlling its contaminants, this work is focused on investigating strategies for controlling the acetaldehyde content in bioethanol. Acetaldehyde is the contaminant responsible for the increase in biofuel acidity during storage time.

2. DESCRIPTION OF PROCESS

A typical industrial installation for bioethanol production in Brazil, according to MARQUINI et al. (2008), is shown in Fig. 1. This industrial installation is composed by 3 columns, two stripping ones (A and B1) and the rectifying column B. Column A, a equipment for wine stripping, is composed by 22 plates, 1 reboiler and no condenser. These plates have Murphree efficiency of 0.65, the total pressure drop of this column is 18437 Pa, the pressure of stage 1 is 138932 Pa and the reboiler pressure 157369 Pa. The wine or beer, industrial denominations of the fermented sugar cane must, is represented by the standard solution given in Table 1. This mixture is fed into the top of column A. The stream named PHLEGM, a vapor stream with ethanol concentration within the range 35-45 mass%, is fed into the bottom of column B. STILLAGE and WHITE STILLAGE, streams withdrawn from the bottoms of columns A and B1, respectively, must have an ethanol content not larger than 0.02 mass%.

Column B, the phlegm rectification column, is composed by 45 plates plus a condenser, has Murphree efficiency of 0.50, a total pressure drop of 38932 Pa, condenser pressure of

100000 Pa and bottom stage pressure of 138932 Pa. Bioethanol is extracted as top product of column B with 93 mass% of ethanol. Column B1, the phlegm stripping column, is fed with the bottom product of column B. This column is composed by 18 plates plus a reboiler, has Murphree efficiency of 0.60, total pressure drop of 8042 Pa, and the reboiler pressure equal to 146974 Pa.

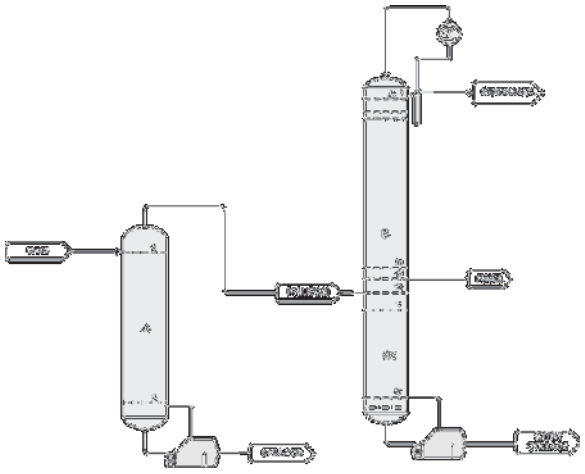


Fig. 1 - Brazilian Bioethanol Industrial Plant

Table 1. Typical composition of industrial wine used in the simulations.

Component	Concentration (mass fraction)	Reference
Water	0.93495357	By difference.
Ethanol	6.450×10^{-2}	Oliveira (2001)
Methanol	3.200×10^{-7}	Boscolo et al. (2000)
Isopropanol	1.020×10^{-6}	Cardoso et al. (2003)
Propanol	3.000×10^{-5}	Oliveira (2001)
Isobutanol	2.780×10^{-5}	Oliveira (2001)
Isoamyl alcohol	4.250×10^{-5}	Oliveira (2001)
Ethyl Acetate	7.690×10^{-6}	Oliveira (2001)
Acetaldehyde	2.000×10^{-6}	Oliveira (2001)
Acetic Acid	4.351×10^{-4}	Oliveira (2001)

3. MATERIALS AND METHOD

The first part of the present work focused on the steady-state simulation of a typical industrial unit, such as that shown in Fig. 1. The simulations were conducted using the commercial software Aspen Plus, by Aspen Tech, and aimed to investigate the operation of the industrial system by analyzing the effects of operational conditions upon the concentration profiles in columns A, B and B1. The second part was conducted using the module Aspen Dynamic, by Aspen Tech, so that some control strategies could be tested in order to keep the acetaldehyde level in bioethanol within the required limits. In this way the acidity increase of the biofuel during storage period could be prevented. The package RADFRAC for simulating distillation columns within Aspen Plus was selected in order to represent the whole industrial system. This package uses a rigorous method of calculation for solving the set of balance and equilibrium equations based on the MESH system described in detail by KISTER (1992). According to a detailed and rigorous analysis (Meirelles et al., 2008), previously performed for the vapor-liquid equilibrium of the binary mixtures formed by the wine components (Table 1), the NRTL model and a corresponding set of parameters were selected for representing the liquid phase non-ideality and the Virial equation, together with the approach based on HAYDEN-O'CONNELL (1975), for estimating the vapor phase fugacities.

Wine was fed into column A (see Fig. 1) with a mass flow of 202542 kg/h, at 94 ºC and the composition given in Table 1. The ethanol concentration in the bottom product of column A was fixed in 200 mg/kg (0.02 mass %) and the mass flow of bioethanol was varied around 14000 kg/h with at least 93 mass% of ethanol, corresponding to an approximately daily production of 465 m^3. In the bottom of column B1 the ethanol concentration was not fixed but it level was ever less than 200 mg/kg. In accordance with industrial information, the fusel stream mass flow was fixed in 41 kg/h, almost 0.3% of the bioethanol mass flow. Reflux and bioethanol stream mass flows were varied and the corresponding concentration profiles investigated.

For the dynamic simulation, in a first step a PID controller was used with the aim of controlling the acetaldehyde content (controller variable) in bioethanol, by manipulating the reflux stream and bioethanol mass flows (manipulated variables), after a perturbation in acetaldehyde concentration was imposed to the feed stream (wine). In a second step, the degassing system was tested to control the acetaldehyde content in bioethanol.

The degassing system is based on the association of two or more partial condensers in the top of column B. The vapor stream of each partial condenser is fed into the next one and the liquid streams return to the top of the column. In the last condenser, a small amount of vapor phase is withdrawn as a DEGASSING stream. According to the maximum level of allowed acetaldehyde contamination, the temperature of the last condenser can be varied and more or less mass of degassing can be generated.

4. RESULTS AND DISCUSSION

Almost all bioethanol fed into column A was stripped from the liquid phase and transferred via the PHLEGMA stream to column B. Except for acetic acid, all congeners (minor components in wine) are concentrated in the PHLEGMA stream and also transferred to column B. Fig. 2 shows the concentration profiles of water and ethanol along columns B (stages 1, condenser, to 46) and B1 (stages 47 to 65, reboiler). An alcoholic graduation of 93.0 mass% was obtained. Note that this value is within the concentration range required by the Brazilian legislation for hydrous bioethanol (Table 2).

Fig. 3 shows the concentration profiles for high alcohols. High alcohols, containing mainly isoamyl alcohol, are extracted from column B as a side stream named FUSEL stream.

Fig. 4 shows the concentration profile for acetaldehyde and acetic acid in columns B and B1. Acetaldehyde profile

indicates that this contaminant is concentrated in the biofuel stream.

ANP, the Brazilian National Petroleum Agency, is the public institution responsible for setting quality standards for fuels and biofuels. Copersucar, one of the largest Brazilian trading companies for sugar and bioethanol export, also sets specific quality standards according to the requirements of its clients. Table 2 shows the main specifications for bioethanol according to ANP (AEHC) and Copersucar (H1 and H2), and also some of the results obtained by steady-state simulation of the industrial plant (SIM). According to the simulation results the bioethanol produced fulfil the requirements of the Brazilian legislation and even most of the requirements set by Copersucar.

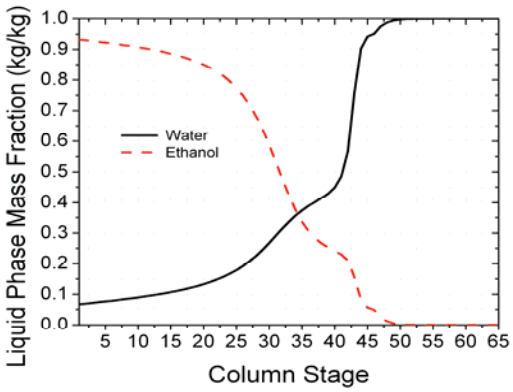


Fig. 2. Concentrations profile of ethanol and water in columns B (stages 1-46) and B1 (stages 47-65).

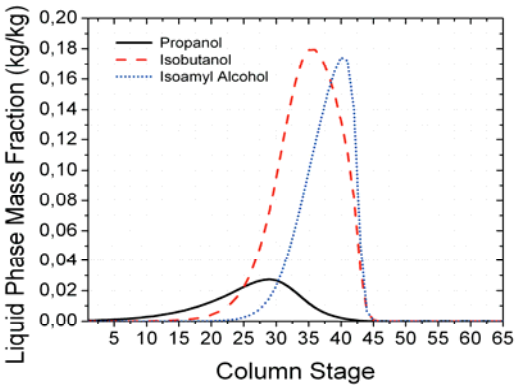


Fig. 3. High alcohols profiles in columns B (stages 1-46) and B1 (stages 47-65).

Acetaldehyde concentration is not a quality parameter fixed by ANP for the biofuel (Table 2). In case of the simulation results, the obtained acidity values, were far below the limit set by the Brazilian legislation. However, during the storage period acetaldehyde can oxidize to acetic acid and deteriorate the biofuel quality, increasing its acidity. If all acetaldehyde content present in the simulated fuel ethanol (Table 2) oxidizes to acetic acid, the product acidity would be increased to 33.5 mg/L. With this value, the biofuel would be outside the standards qualities established by the Brazilian legislation (Table 2). For this reason, the concentration of acetaldehyde in biofusel must be strictly controlled to prevent that the acidity level exceeds the legislation limits along the storage time. On the other hand, Brazil is nowadays the largest bioethanol exporter and the use of this bioproduct is increasing worldwide not only as an alternative energy source as well as an input material for chemical, pharmaceutical, perfume and beverage industries. Although these other uses may require further purification steps, sometimes conducted at the importing country, the Brazilian exporters are opting for defining stricter quality standards, such as the values specified by Copersucar (see Table 2). This highlights the importance of monitoring and controlling the contamination levels of minor components, such as acetaldehyde and high alcohols, in bioethanol.

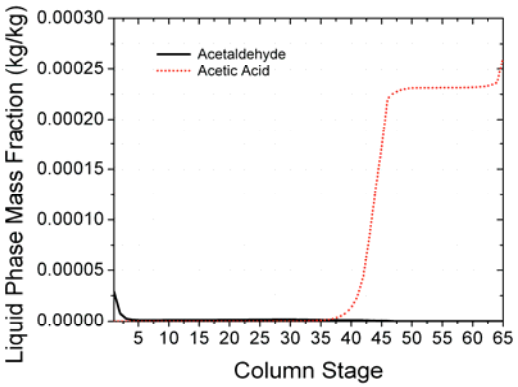


Fig. 4. Acetaldehyde and acetic acid profiles in columns B (stage 1-46) and B1 (stages 47-65)

Table 2. Bioethanol quality standards, ANP (AEHC), Copersucar (H1 and H2) and the simulation results (SIM).

Spec.	Unities	Bioethanol			
		AEHC	H1	H2	SIM
Alcoholic Graduation	mass%	92.6-93.8	≥ 92.8	≥ 93.8	93.2
Acidity (Acetic Ac.)	mg/L	≤ 30	≤ 20	≤ 10	Trace
Density (20ºC)	kg/m^3	807.6-811.0	-	-	807.1
Acetaldehyde	mg/L	-	≤ 50	≤ 10	24.6
High Alcohols	mg/L	-	≤ 400	≤ 50	332.5

Data on the mechanism and kinetics of acetaldehyde oxidation to acetic acid can be found in WANG et al. (1992) and XU et al. (2000). In order to avoid the risk of this oxidation during biofuel storage one of the possible strategies is to reduce acetaldehyde content in biofuel to a minimal value. In the second part of this work, some strategies to control the acetaldehyde content were investigated. All the strategies were based in a PID loop control, with the aim of keeping acetaldehyde concentration in bioethanol constant even if a perturbation increases its content in the wine. Figure 5 shows the simplest configuration of column simulated in the present work. As acetaldehyde is a very light component, the total amount of this substance present in the wine will

contaminate bioethanol if this configuration is used. For this reason no control strategy would be able to avoid an increase of acetaldehyde contamination in bioethanol in case of a slight increase in its concentration in the wine. In fact, attempts to avoid this contamination, by using reflux and/or bioethanol flow, according to the loop control represented in Fig. 5, failed. Thus two alternative solutions are suggested and they include changes in the industrial installation.

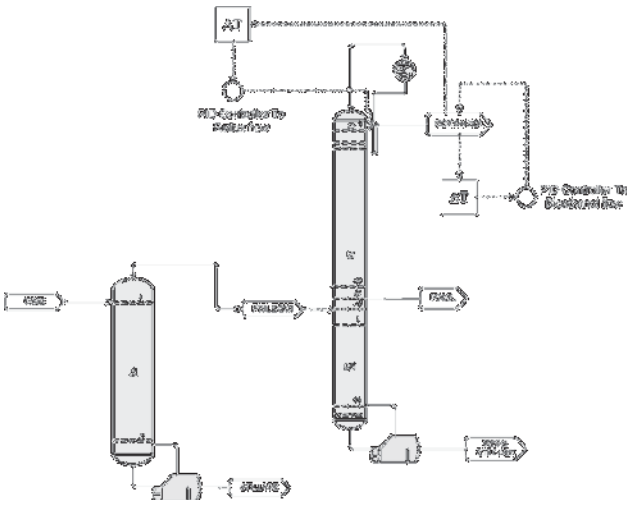


Fig. 5. Loop control for acetaldehyde concentration in bioethanol

The first alternative installation includes a degassing system, as that shown in Fig. 6 and explained above. Such a system makes easier the control of acetaldehyde content in bioethanol. As a very light component, acetaldehyde concentrates in the vapor streams and is eliminated by the DEGASSING stream. Controlling the DEGASSING flow makes possible to eliminate part of the acetaldehyde contamination, although this also causes small losses of the bioproduct.

Fig. 7 shows steady-state results for DEGASSING flow, ethanol mass flow in degassing and acetaldehyde content in bioethanol as a function of the last condenser temperature. The increase of this temperature increases the degassing flow, and by consequence increases the mass flow of ethanol in degassing stream, and decreases the acetaldehyde concentration in the bioethanol. These results show that the control of the temperature of the last condenser in the degassing system can control the concentration of acetaldehyde in the bioethanol. Taking this into account, a simple PID controller was developed to control the temperature of the last condenser of the degassing system (see Fig. 6). In this loop control, the controller variable was the acetaldehyde content in bioethanol and the manipulated variable was the temperature of the last condenser. The stack point (maximum level of the acetaldehyde in bioethanol) was fixed in 25.3 ppm (2.530×10^{-5} kg/kg). With this concentration, even if all the acetaldehyde oxidize to acetic acid, the mass of acid formed will not be sufficient to exceed the acidity maximum level fixed by ANP (Table 2). In order to better represent the industrial process, carbon dioxide (CO_2) produced during fermentation was included in the wine composition in a concentration of 0.0011 kg/kg. This value was determined assuming that the alcoholic fermentation industrial process is performed in closed vat with light over pressure (600 to 800 mm of water) and temperatures close to 35 °C. Considering that gas phase inside the vat is composed of saturated CO_2 with vapors of ethanol and water, the NRTL model and the Henry constant for CO_2 (Dalmolin et al., 2006) was used in order to estimate the solubility of CO_2 in the wine. The estimated values varied within the range 1050 to 1150 mg/kg. The acetaldehyde concentration in the wine was increased to 2.100×10^{-6} kg/kg and after 3 hours decreased to 1.900×10^{-6} kg/kg, in order to demonstrate the efficiency of the degassing system. The concentration of the other wine components were kept constant in the values indicated in Table 1, except for water whose value was appropriately adjusted. The results are present in the Fig. 8.

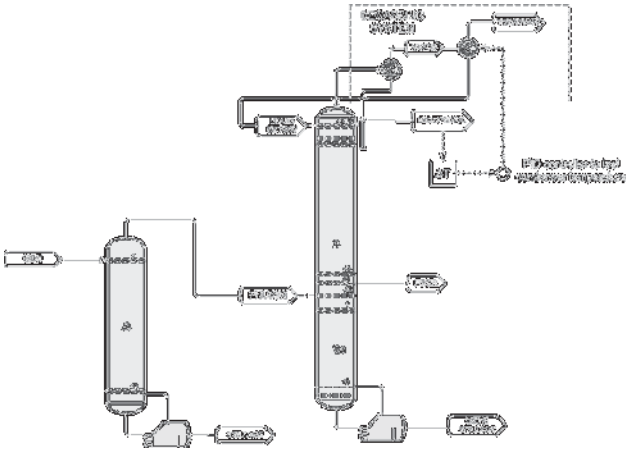


Fig. 6. Industrial plant with degassing system

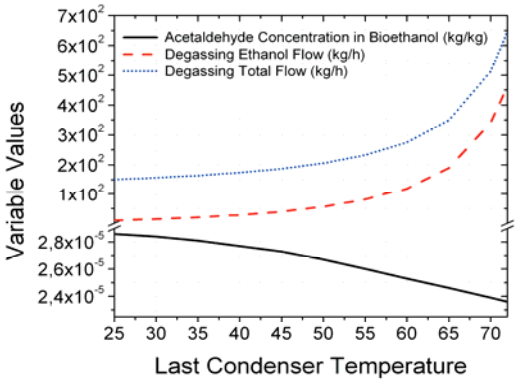


Fig. 7. Acetaldehyde content and degassing flow as a function of last condenser temperature

As is possible to observe in Fig. 8, the control system based in a PID controller has a good performance in avoiding a contamination of acetaldehyde in bioethanol. A direct dependence between the controller variable (biofuel acetaldehyde concentration) and the manipulated variable (last condenser temperature) was observed. In case of an

increase of acetaldehyde concentration in the wine the PID controller increases the last condenser temperature and, in consequence, a large degassing flow is withdrawn of the equipment. The acetaldehyde level in bioethanol reaches safe values after 40 minutes and stabilizes after one hour. The reverse process occurs when the concentration of acetaldehyde in wine is decreased (see Fig. 8).

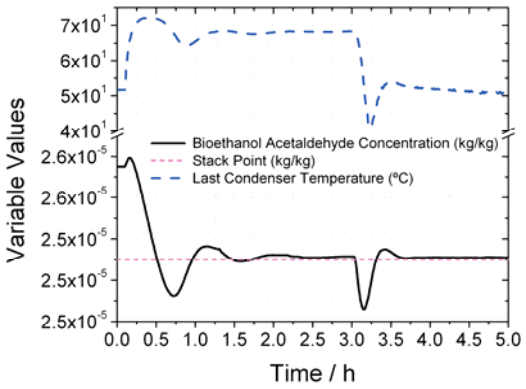


Fig. 8. Results of PID controller in degassing system (industrial installation)

Despite this good performance, the configuration with a degassing system may exhibit some difficulties in case of a large wine contamination with acetaldehyde. Large concentrations of acetaldehyde in wine require larger flow of degassing stream in order to reduce the biofuel contamination. A larger degassing mass flow increases ethanol losses (see Fig. 7). Therefore, the total loss of the ethanol in the production system can reach levels higher than those accepted by industry. An alternative configuration better for a wine with larger acetaldehyde contamination is the pasteurized bioethanol installation shown in Fig. 9. In this kind of installation two news columns (D and A1) are added to the original system. These columns concentrate the major part of wine volatile compounds, including acetaldehyde, and eliminate part of them via the SECOND ALCOHOL stream withdrawn from the top of column D.

In column B bioethanol is withdrawn from a tray close to the column top. In the top of Column B a further SECOND ALCOHOL stream is also withdrawn. According to Fig. 4 acetaldehyde is concentrated in the trays located close to the top of column B. For this reason streams such as the two SECOND ALCOHOL ones are concentrated in acetaldehyde and other light minor components, for instance ethyl acetate. These contaminants are taken away by the top streams and bioethanol, withdrawn from column B as a side stream, has its acetaldehyde content decreased. On the other hand, small amounts of ethanol are not recovered as the main product (bioethanol), being extracted in those byproduct streams. Such scheme is more appropriate for producing bioethanol from a wine with larger contamination of light components or in case the bioproduct must have a higher purity.

Fig. 10 shows the results of steady state simulations performed for the pasteurized bioethanol installation. For this simulation acetaldehyde concentration in the wine was increased to approximately 10 times the value of the previous simulations (new concentration equal to 1.900×10^{-5} kg/kg), representing a larger contamination, closer to the industrial wine, according to Oliveira (2001).

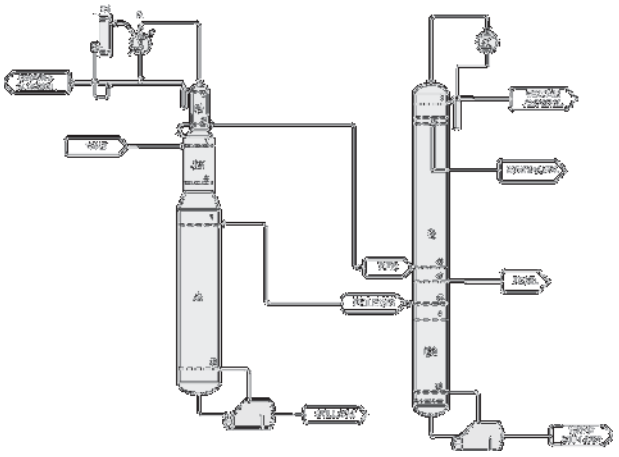


Fig. 9. Industrial plant for bioethanol with second alcohol streams.

The main objective of those simulations was to show that, varying the mass flow of the second alcohol stream in column B, it is possible to reduce considerably the concentration of acetaldehyde in bioethanol. According to Fig. 10, the increase of the mass flow of the second alcohol stream reduces acetaldehyde contamination without influencing, in a significant way, the bioproduct alcoholic graduation. In these simulations only the second alcohol stream in top of column B was varied, keeping the second alcohol stream in top of column D fixed at the value 400 kg/h.

This means that a relative larger acetaldehyde contamination is contained in the second alcohol stream, a result that makes easier the control of this contamination in the main product (pasteurized bioethanol) by means of the degassing system. For this reason a loop control similar to that of Fig. 6, connecting the acetaldehyde concentration in pasteurized bioethanol (controller variable) to the last condenser temperature (manipulated variable), was tested. The wine acetaldehyde concentration was increased to 2.000×10^{-5} kg/kg and the last partial condenser temperature was varied to stabilize the bioethanol acetaldehyde concentration at 2.450×10^{-5} kg/kg. With this value, the problem of acetaldehyde oxidation during storage time was eliminated. The result of this simulation was presented in Fig. 11.

The results show that in almost 2 hours the acetaldehyde concentration reaches the required value although the stabilisation time is approximately 7 hours. This result suggests that the degassing system is an excellent alternative for acetaldehyde control in bioethanol, provided that the wine contamination with acetaldehyde is not too large.

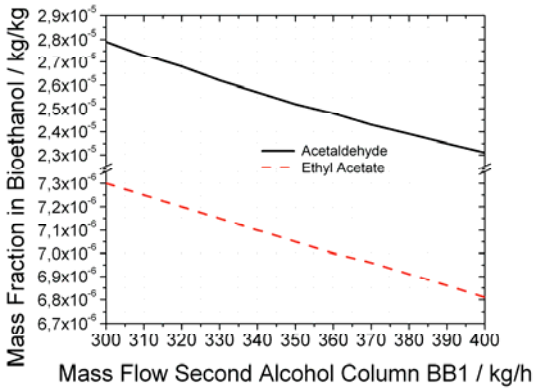


Fig. 10. Volatiles content in bioethanol in function of second alcohol flow of column BB1

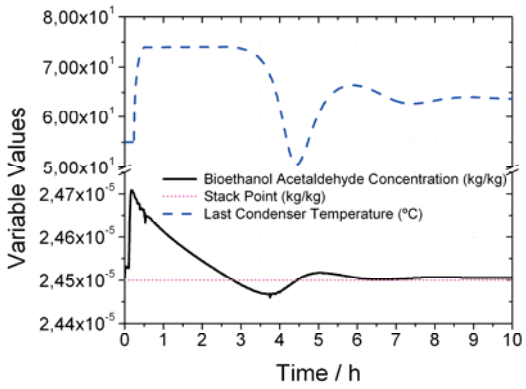


Fig. 11. Results of PID controller in degassing system (pasteurized bioethanol installation)

5. CONCLUSION

Production of bioethanol as a renewable fuel or as an input commodity to be used in other industrial branches requires the reduction and control of several contaminants contained in the fermented must. In the present work special attention was focused on controlling acetaldehyde contamination. Analyzing the results presented it is possible to conclude that the wine (must) acetaldehyde concentration will determine the type of industrial installation and the type of control to be used to regulate the acetaldehyde in bioethanol and prevent problems with its oxidation during storage. Thus, for wine with less than 2.0×10^{-6} kg/kg of acetaldehyde, the industrial installation without degassing system is appropriate. For wine concentrations within the range 2.0×10^{-6} to 2.2×10^{-6} kg/kg, the degassing system is required. In case of wine concentrations within the range 2.2×10^{-6} to 2.0×10^{-5} kg/kg, the pasteurized bioethanol installation is the most appropriate one. For concentrations within the range 2.0×10^{-5} to 2.2×10^{-5} kg/kg the degassing system should be included in the pasteurized bioethanol installation. Finally, for musts with higher acetaldehyde concentration ($\geq 2.2\times10^{-5}$ kg/kg) the pasteurized bioethanol installation with a PID controller to regulate the mass flow of second alcohol is probably the best way to prevent problems with acetaldehyde oxidation during storage.

ACKNOWLEDGMENTS

The authors acknowledge the financial support of FAPESP (05/53095-2 + 08/56258-8), CNPq (306250/2007-1) and CAPES.

REFERENCES

Boscolo, M., Bezerra, C.W.B., Cardoso, D.R., Neto, B.S.L., and Franco, D.W. (2000). Identification and Dosage by HRGC of Minor Alcohols dnd Esters in Brazilian Sugar-Cane Spirit. *Journal of The Brazilian Chemical Society*. 11(1), 86.

Cardoso, D.R., Lima-Neto, B.S., Franco, D.W., Nascimento, R.F. (2003). Influência do Material do Destilador na Composição Química das Aguardentes de Cana – Parte II. *Química Nova*, 26(2), 165.

Dalmolin, I., Skovroinski, E., Biasi, A., Corazza, M.L., Dariva, C., Oliveira, J. V. (2006). Solubility of carbon dioxide in binary and ternary mixtures with ethanol and water. *Fluid Phase Equilibria*. 245, 193-200.

Decloux, M. and Coustel, J. (2005). Simulation of a neutral spirit production plant using beer distillation. *International Sugar Journal*, 107 (1283), 628-643.

Gaiser, M., Bell, G. M., Lima, W., Roberts, N. A., Faraday, D. B. F., Schulz, R. A., Grob, R. (2002). Computer simulation of a continuous whisky still. *Journal of Food Engineering*, 51(1), p. 27-31.

Hayden, J. G. and O'Connell, J. P. (1975) A Generalized Method for Predicting Second Virial Coefficients. *Ind. Eng. Chem.. Process Des. Dev*., V 14(3), 209-216.

Kister. Henry Z. (1992). *Distillation Design*. United States: McGraw-Hill. Inc. 709 p.

Marquini, M.F., Maciel Filho, R., dos Santos, O.A.A., Meirelles, A.J.A., Jorge, L.M.M. (2008). Reduction of Energy Comsumption and Effluent Generation in Ethanol Distilleries., 09/2008, *XVII Brazilian Congress on Chemical Engineering - COBEQ - 2008*,Vol. 1, pp.1-3, Recife, PE, Brazil.

Meirelles, Antonio J.A., Batista, E.A.C., Scanavini, H.F.A., Batista, Fabio R.M., Ceriani, R. Distillation Applied to the Processing of Spirits and Aromas. In: M. Angela A. Meireles (Ed.). *Extracting Bioactive Compounds: Theory and Applications*. New York: CRC Press, 2009. Chapter 3, 75-136.

Oliveira, E. S. (2001). PhD thesis (in potuguese), Faculty of Food Engineering, University of Campinas, Campinas, Brazil.

Wang, S.Q., Zhang, R.F., Wang, J.C. (1992). Mathematical model of the process for the oxidation of acetaldehyde to acetic acid. *Computers in Industry*, 18 (2), 213-219.

Xu, L., Boring, E., Hill, C.L. (2000). Polyoxometalate-Modified Fabrics: New Catalytic Materials for Low-Temperature Aerobic Oxidation. *Journal of Catalysis*, 195(2), 394-405.

Process Monitoring and Diagnosis

Poster Session

Sensor fault detection and isolation for single, multiples and simultaneous faults: Application to a waste water treatment process

Fragkoulis D. * **, Roux G. * ** and Dahhou B. * **

* LAAS-CNRS ; Université de Toulouse ;
7, Avenue du Colonel Roche, F-31077 Toulouse, France (e-mail: roux@laas.fr).
** Université de Toulouse;

Abstract: In this paper, sensor fault detection, isolation and identification model-based approach is designed. We introduce a new state variable so that an augmented system can be constructed to treat sensor faults as actuator faults. The approach uses the model of the system and a bank of adaptive observers to generate residuals. Structured residuals are defined in such way to isolate the faulty sensor after detecting the fault occurrence. The advantage of this method is that we can treat single, multiple and simultaneous sensor faults. In this study, we consider that only abrupt faults in the system sensor can occur. The proposed strategy is validated by simulation results of a nonlinear model of a waste water treatment process.

1. INTRODUCTION

In all industrial processes the reliability and the security of the system is a very important task. A fault may occur in all possible location, such as actuators, sensors and system's parameters. Fault detection techniques could prevent from all the undesirable consequences. In order to improve efficiency, the reliability can be achieved by fault-tolerant control, which relies on early fault detection, using fault detection and isolation (FDI) procedures. So FDI is becoming an attractive topic. Model based fault detection and diagnosis systems have found extensive use because of the fast response to abrupt failure and the implementation of the model based FDI in real-time algorithms. A comprehensive review of the different methods for FDI and their applicability to a given physical system has been presented in ([Iserman, 1994] and [Venkatasubramanian *et al.* 2003]). A variety of effective methods can be used to realize FDI, such as differential geometric approach [De Persis and Isidori 2001], sliding mode observer ([Edwards and Spurgeon 1994] and [Xing-Gang and Edwards 2005]), and adaptive control technique ([Frank, 1994] and [Hammouri *et al.* 1999]).

The progressive deterioration of the water resources and the great quantity of polluted water produced in the industrialized companies, give to the waste water treatments (WWT) a great importance in the safeguarding of water quality. The new directives and regulations (the directing 91/271/CEE referring itself to the European countries) impose the adoption of specific indices for the quality of treated waste water. Taking into account the current ecological problems, it is realistic to believe that this tendency will continue. At the same time, the existing factories increase thanks to the growth of the urban sectors and this situation requires more effective treatments of the used water. Consequently, we want that such an industry, almost always, operates with the maximum effectiveness.

Generally, the recent evolution of the legislation of some countries, about the use of surface or subsoil waters, is such that the total reuse of the water used in the processes became a very important issue. So the waste water treatment became a part of the production process, where the quality control of effluent is very important. Since the weak operation of the treatment can carry out to an important loss of production and to ecological problems.

The paper is organized as follows. In Section 2, we present the class of the nonlinear systems that we study, the filter that we apply to form the new extended system and the formulation of the fault problem. Then we give the principle of the fault detection and isolation scheme and the synthesis of the observer. Section 3 describes the waste water treatment process which is used to show the effectiveness of the proposed method. In Section 4, we give simulation results that illustrate the method for single, multiple and simultaneous faults. Conclusion and perspectives end the paper.

2. EXTENTED MODEL AND PROPOSED METHOD

2.1 Filter for the system's output

We consider the following class of nonlinear systems:

$$\begin{cases} \dot{x} = f(x) + g(x)u \\ y = Cx \end{cases} \qquad (1)$$

where $f(x)$ is a nonlinear vector function from $\Re^n$ to $\Re^n$, $g(x) \in \Re^{n\times m}$ is a matrix function whose elements are nonlinear functions, $C \in \Re^{p\times n}$ is a matrix, $u \in \Re^m$ is the input vector and $y \in \Re^p$ is the output vector. Throughout

this paper, we assume that only constant sensor faults can occur $y_j^f(t) = y_j(t) + f_{sj}$, that is $y_j^f \equiv \theta_j$ for $t \geq t_f$, $j \in 1, 2, \cdots p$, and $\lim_{t\to\infty} | y_j(t) - \theta_j | \neq 0$, where θ_j is a constant and $y_j^f(t)$ is the actual output of the j^{th} sensor when it is faulty, while $y_j(t)$ is the expected output when it is healthy.

In [Chee and Edwards 2003] the authors presents a method for the linear system where the output vector passes through two orthogonal matrices $T_{r,1}$ and $T_{r,2}$. At the same time these matrices make the separation of the outputs at $y_1 \in \Re^{p-h}$ and $y_2 \in \Re^h$ where y_1 are the outputs without fault and y_2 are the outputs with a fault. The same manipulation of the outputs for the nonlinear system (1) is impossible but there is a similar method proposed in [Chen and Saif 2006] for the class of the nonlinear systems (1) that we presented above. We will apply to the output vector y a filter of the form:

$$\dot{\xi} = A_f \xi + B_f y \quad (2)$$

Where the state vector is $\xi \in \Re^p$, we select $A_f \in \Re^{p\times p}$ as a Hurwitz matrix and $B_f \in \Re^{p\times p}$ is chosen as an invertible matrix. We form the new input $w = \begin{bmatrix} u \\ y \end{bmatrix}$ and we define the extended system of the form:

$$\dot{z} = \underline{f}(x, \xi) + \underline{g}(x) w \qquad (3)$$

Where the vector $z \in \Re^{n+p}$ is the new state $z = \begin{bmatrix} x \\ \xi \end{bmatrix}$, $\underline{f}(x, \xi) \in \Re^{n+p}$ is a vector with nonlinear and linear elements ($\underline{f}(x, \xi) = \begin{bmatrix} f(x) \\ A_f \xi \end{bmatrix}$). The matrix $\underline{g}(x) \in \Re^{(n+p)\times(m+p)}$ is a matrix with nonlinear and linear elements ($\underline{g}(x) = \begin{bmatrix} g(x) & 0_{n\times p} \\ 0_{p\times m} & B_f \end{bmatrix}$) and finally the vector $w \in \Re^{m+p}$ is the new input vector. So, as we have seen with this transformation we have extended the system and the initial sensor fault problem has become, after the transformation, an actuator fault problem. The output vector y of the system has become a part of the input vector w of the new system. Based on the approach developed in [Blanke *et al.* 2003], it is easy to build the corresponding extended faulty model:

$$\begin{cases} \dot{z} = \underline{f}(x, \xi) + \sum_{j\neq l} \underline{g}_j(x) w_j + \underline{g}_l(x) \theta_l \\ y = \begin{bmatrix} C & 0_{p\times p} \end{bmatrix} z \end{cases} \qquad (4)$$

where we have a fault in the l^{th} actuator and $\underline{g}(x) = [\underline{g}_1(x) \cdots \underline{g}_{m+p}(x)]$.

The new system input as we already mentioned is the vector *w*. This vector includes the inputs and the outputs of the system (1), $w^T = [u_1 \ \cdots \ u_m \ | \ y_1 \ \cdots \ y_p]$. In this paper, we are focusing only in sensors faults. As the method that it will be used is an actuator fault detection and isolation method, the inputs of the new vector *w* that we are interested are from w_{m+1} to w_{m+p}.

2.2 The fault detection and isolation scheme

After this transformation, the problem has become an actuator fault detection and isolation problem where the faults have the same properties with the ones presented in the begin of the subsection *2.1*; only that in the place of the output y we have the input w. For the fault detection and isolation, we will develop a bank of p adaptive observers, where $\hat{\theta}$ is the fault estimation [Chen and Saif 2005]. The form of the adaptive observer that we will use in this bank for the l^{th} actuator is:

$$\begin{cases} \dot{\hat{z}}_l = \underline{f}(x, \xi) + \sum_{j\neq l} \underline{g}_j(x) w_j + \underline{g}_l(x) \hat{\theta}_l + H_l(\hat{z} - z) \\ \dot{\hat{\theta}}_l = -2\gamma (\hat{z}_l - z)^T P_l \underline{g}_l(x) \quad m+1 \leq l \leq m+p \end{cases} \qquad (5)$$

Where H is a Hurwitz matrix that it can be chosen freely, γ is a design constant and P is a positive definite matrix. We can calculate the matrix P and H with the help of the following Lyapunov equation:

$$H^T P + PH = -Q \qquad (6)$$

where Q is a positive definite matrix that it can be chosen freely. The analysis of the method can be found in [Chen and Saif 2005] along with all the proofs and details. An application of this method for an actuator fault detection and isolation to the same system that we studied can be found in [Fragkoulis *et al.* 2007]. The residual r_i that it is proposed in this paper is the difference between the estimation of the fault $\hat{\theta}_i$ determined in (5) and the output of the system so:

$$r_i = \hat{\theta}_{m+i} - y_i, i \in [1 \ldots p] \quad (7)$$

The residuals are designed to be sensitive to a fault that comes from a specific sensor and as insensitive as possible to all the others sensor faults. This residual will permit us to treat not only with single faults but also with multiple and simultaneous faults. To facilitate the isolation of the fault the structured residual will be used, and in particular the Boolean method introduced in [Gertler 1998] with simple thresholds δ_{si}.

$$\varepsilon_i(t) = \begin{cases} 1 & if \quad |r_i(t)| \geq \delta_{si} \\ 0 & if \quad |r_i(t)| < \delta_{si} \end{cases} \qquad (8)$$

$$\Phi = [\varepsilon_1(t)\ \varepsilon_2(t) \ldots \varepsilon_p(t)] \quad (9)$$

$$r_s \leftarrow \Phi f_s \qquad (10)$$

So the five steps for this new FDI scheme are:

1. we determine the filter as in (2) for the augmented space.
2. we form the new faulty model (4) and the new input vector *w*.
3. we build a bank of p observers as in (5) for the detection and isolation of the fault.
4. we generate the residuals r_i (7).
5. from the thresholds δ_{si} we elaborate the structural matrix Φ and then
6. from (10) we generate the structured residuals r_s for the fault isolation and identification.

3. WASTE WATER TREATMENT PROCES MODEL

The process of water treatment by activated sludge, invented in Manchester in 1914, industrially reproduced the purifying effect of the rivers, and became the principal current process of purification. It consists of an aerobic biological system in which the biological floc (biofloc) are continuously recycled and given in contact with organic waste water in the presence of oxygen. Oxygen is usually provided by bubbles of air, insufflated in the mixture of liquid and sludge under conditions of turbulence or by units of surface mechanics or by other aeration types.

A plant of water purification with activated sludge generally consists of a system of treatment in two phases (figure 1). The first phase of the treatment consists in eliminating pollutant in suspension, which mainly includes the degreasing, the de-sanding and the de-oiling. Now, we present the second phase which can be described by three reactors placed in cascade. The first reactor called primary decanter receives polluted water coming from the urban or industrial environments. Water penetrates then in a second reactor, called aerated basin, which constitutes the heart of the plant. The treatment is based on setting in contact of a bacterial population (micro-organisms) with organic matter contained in the effluent to treat. In the aerated basin occur initially a fast adsorption and flocculation of the colloidal matters in suspension and of the organic matter soluble by the activated sludge. Then there is a progressive oxidation of a synthesis of the adsorbed organic matter and of the extracted organic matter. Finally, water undergoes a last treatment in the third reactor, called settling tank. This one delivers purified water after the decantation of sludge. A part of this latter is recycled in the aerated basin (recycled sludge) and sludge in excess is evacuated for a suitable external treatment.

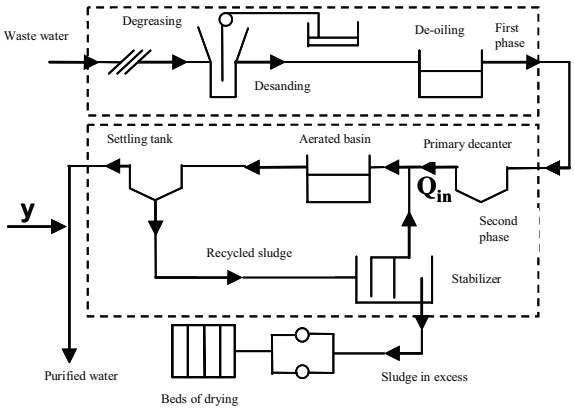


Figure.1 Waste water treatment process

The mathematical model for the activated sludge process (aerated basin and settling tank) is based on the equations, resulting from mass balance considerations, carried out on each of the reactant of the process.

$$Variation = \pm Conversion + Feeding - Drawing\ off$$

All the details about the system and the values of the model parameters can be found in ([Nejjari 2001] and [Fragkoulis *et al.* 2007]). The FDI scheme will monitor the sensors S_I, S_S, X_I, X_S, X_H and S_O, measuring the output vector y of the settling tank (cf. figure 1), by using a bank of adaptive observers. The algorithm for this model is constituted by a bank of six adaptive observers for the fault detection, isolation and identification. More details about the observer synthesis can be found in [Fragkoulis *et al.* 2008].

4. SIMULATION RESULTS

In this section, we will give the results obtained from the developed method for one or more sensor faults. We have to mention that in the case of multiple and simultaneous faults, while the second fault occurs the first fault still acts in the system. The banks of adaptive observers run simultaneously with the system. The considered installation is a closed loop system. So the presence of the controller makes the sensor fault problem more complex. In this case, the fault affects not only the faulty sensors but also the system's dynamic (the other outputs of the system). The sampling period is one sample per hour, the value of all the constant thresholds are $\delta_{si} = 0.5$. Finally we have to mention that all the outputs and so all the faults are in *mg/l*.

4.1 Single fault

We have applied a fault with magnitude $f_{s5} = -2.2\, mg/l$ at time $t = 50$ days in the fifth sensor X_H. In Figure 2, we present the six residuals r_i associated to the six observers. The six residuals in the begin needs a short time period to

converge. This time depends on the initialisation time of the observer's bank, so as to be ready for a fault detection and isolation. After, they reach a constant value and stays there until the fault occurrence.

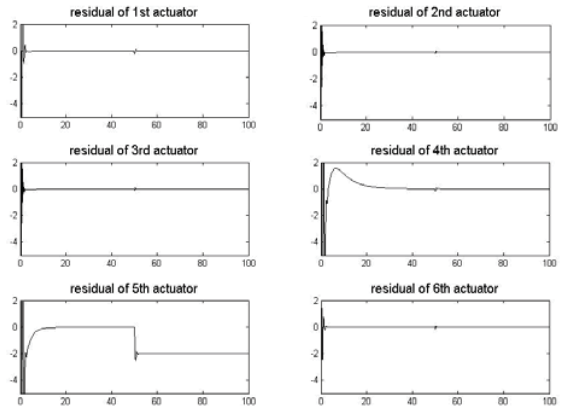


Figure.2 Residuals r_i for a single fault

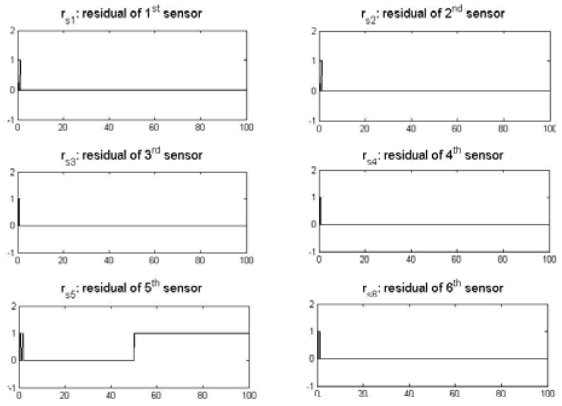


Figure.3 Structured residuals r_{si} for a single fault

At time $t = 50$ days, we can see that the residual of all the six observers leave zero but after a very short period (one day maximum), all of them return to their initial values, except from the residual associated to the fifth observer that corresponds to the output X_H that it takes a new constant value and remains there. In figure 3 we present the structured residuals for this fault. As expected all the residuals stay at zero except from the residual r_{s5} associated to the fifth sensor that at time $t = 51$ days takes and stays at the value "1". Thus, this fact indicates that this is the faulty sensor. Therefore, we isolate the faulty sensor correctly and rapidly enough. As we already mention, this method not only isolates the fault but also identify its value, which can be used for the system reconfiguration. In this case, the actual value of the fault is $f_{s5} = -2.2\, mg/l$ and the estimated value is $\hat{f}_{s5} = -2.1\, mg/l$, so we had identified the fault very accurately.

4.2 Multiple faults

We have applied a constant fault with magnitude $f_{s3} = -5\, mg/l$ at time $t = 50$ days in the third sensor X_I and one with magnitude $f_{s1} = -3\, mg/l$ in the first sensor S_I at time $t = 60$ days. The fault at the third sensor is still occurred when the fault at first sensor has been introduced. Figure 4 shows, the six residuals associated to the observers, where after the initialisation, they have a constant value until $t = 50$ days. There, all the residuals leave their initial values and only the residual associated to the third observer that corresponds at the third sensor stays to the new value. The other five residuals return to their initial values.

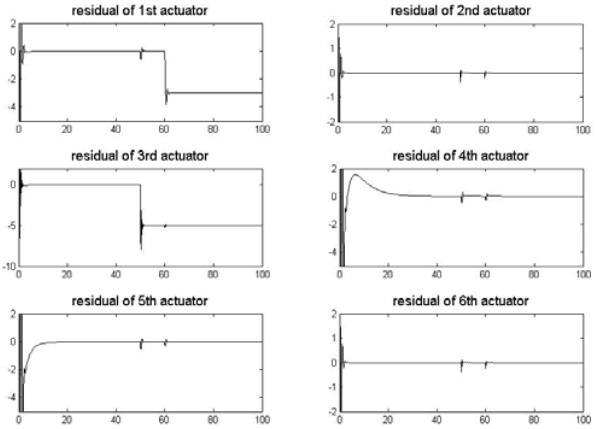


Figure.4 Residuals r_i for multiple faults

At time $t = 60$ days, where the second fault has been entered, the residual that corresponds to the first sensor S_I change from his initial value. It stays at the new value but the other five residuals leave their value and returns to them after a short time period. The third residual, which corresponds to the third sensor where the first fault still occurs, has not been affected by the new fault.

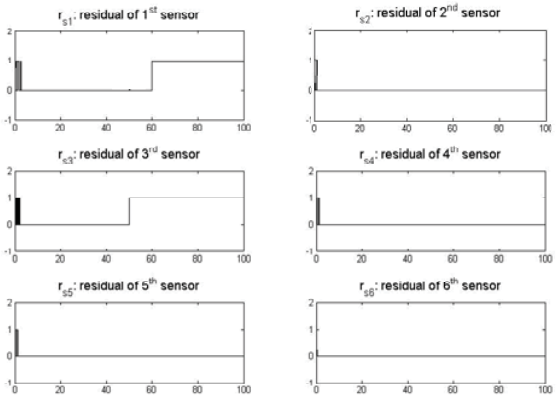


Figure.5 Structured residuals r_{si} for multiple faults

In figure 5, we can see the structured residuals where only the residuals r_{s3} and r_{s1} at time $t = 51$ days and $t = 61$ days respectively leave zero and stay at their new value "1". More generally each fault affects only the corresponding residual and the isolation of the multiple faults has been done. For the fault identification we have: the estimation of the first fault is $\hat{f}_{s3} = -4.5\, mg/l$ and the actual value is $f_{s3} = -5\, mg/l$; the estimation of the second fault is $\hat{f}_{s1} = -2.85\, mg/l$ and the actual value is $f_{s1} = -3\, mg/l$. So we have a good estimation of the fault, not only for the first one where we have a single fault but also for the second one, the multiple fault case.

4.3 Simultaneous faults

We illustrate the case where more than one faults occur at the same time on the system or briefly the simultaneous faults. We have applied two faults: one on the fourth sensor X_S with magnitude $f_{s4} = -13\, mg/l$ and one on the sixth sensor S_O with magnitude $f_{s6} = -2\, mg/l$ at the same time $t = 50$ days.

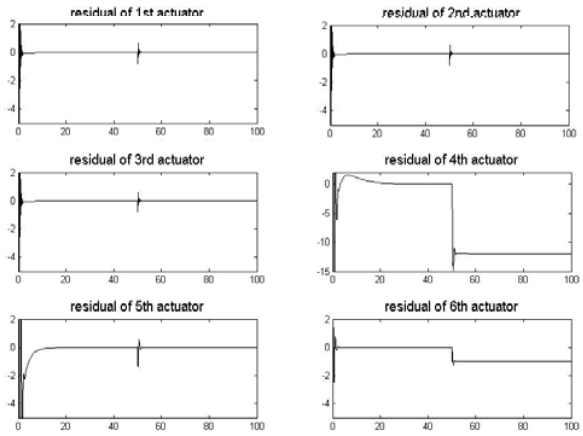


Figure.6 Residuals r_i for simultaneous faults

In Figure 6, we give the residuals associated to the observers and we can see that their values are equal to a constant value until $t = 50$ days where the two faults occur on the system. At that time, all the residuals leaves their initial values and only the residual associated to the fourth observer that correspond to the fourth sensor and the residual associated to the sixth observer that correspond to the sixth sensor stays to their new value; the other four residuals returns to their initials values. Figure 7 presents the structured residuals where only the residuals r_{s4} and r_{s6} leaves zero at time $t = 51$ days, therefore we isolate the two faulty sensors. The identification of the two faults is quite accurate, so for the fourth sensor the estimation is $\hat{f}_{s4} = -12.5\, mg/l$ and for the other one the estimation is $\hat{f}_{s6} = -1.8\, mg/l$.

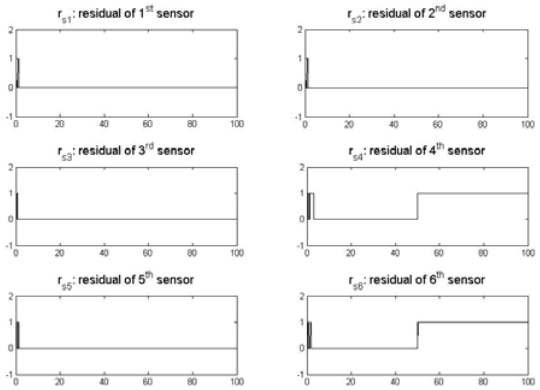


Figure.7 Structured residuals r_{si} for simultaneous faults

4.4 Single fault with real data

We will present the case where the input Q_{in} which is the flow rate input of the aerated basin (cf. figure 1) take his values from a file with real data. These data are collected from a benchmark installed in Terrassa Spain [Nejjari 2001], while the other three inputs have a constant value as in reality. Thus a single fault occurs in one of the six sensors and the method's validity will be presented. In figure 6 we present the input Q_{in} .

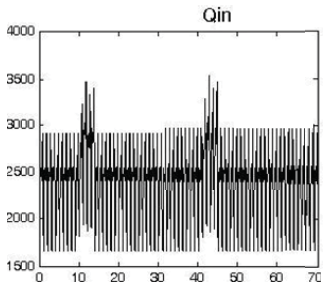


Figure.8 Input Q_{in} with real data

The duration of these data is 70 days and during this time we have two intermittent perturbations, one at the 9th day until the 13th day and another one at the 40th day until the 45th day, caused by the rain. A single fault with magnitude $f_{s4} = -15$ has been occurred at time $t = 50$ days in the fourth sensor X_S .

In figure 9, we show the six residuals associated to the six sensors. As we can see the two perturbations that occurred on the system have a little influence on them. Mainly the first, fourth and fifth residuals have been a little bit affected by them, but the effect can not be misjudged as a fault as long as the residuals remain in the zone defined by the two thresholds δ_{si} . The structured residuals, figure 10, stay at zero during the perturbations.

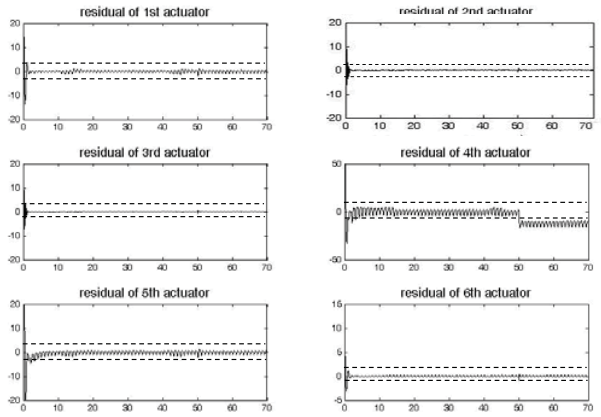


Figure.9 Residuals r_i for single fault with real data

Then at time $t = 50$ days the fourth residual, in both figures, indicates us that there is a fault in the fourth sensor. The simple residual leaves his initial value and gives us the estimation of the fault and the structured residual takes the value "1", so we can easily conclude the source of the fault. For the estimation of the fault we have to use the mean value on a sliding window due to the fact that we have a small oscillation of the value; this mean value is $\hat{f}_{s4} = -16\,mg/l$. In this case the thresholds value is $\delta_{si} = 2$ and they are chosen empirically, also we have to mention that the use of structured residuals facilitates the automatic isolation.

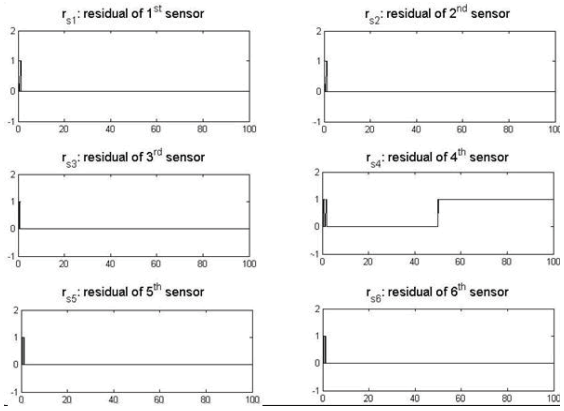


Figure.10 Structured residuals r_{si} for single fault with real data

5. CONCLUSIONS

In this paper, a new method, for sensor fault detection isolation and identification, based on nonlinear observers has been developed. We have reformulated the initial sensor fault problem, by using a transformation filter, to an actuator fault problem. We have designed a known bank of adaptive observers to treat the FDI procedure. Simulation results illustrate the effectiveness of the method for the isolation of single faults, multiple and simultaneous faults. Finally we have validated the proposed method with real data collected from a waste water treatment process benchmark. Our future considerations are to improve the fault estimation in the case of measurement noise by using a better filtering method of the residual. Finally the comparison with the simple method of adaptive observers and mainly the comparison between the isolation time and the fault identification is one of our highly concerns.

ACKNOWLEDGMENTS

The first author would like to thank his sponsor, the Greek State Scholarships Foundation (I.K.Y.), for the financial support.

REFERENCES

Blanke, M., Kinnaert, M., Lunze, J. and Staroswiecki, M. (2003). *Diagnosis and Fault-Tolerant Control*, Springer-Verlag, Berlin.

Chee, P. T. and Edwards, C. (2003). Sliding mode observers for reconstruction of simultaneous actuator and sensor faults, *Proceeding of the 42nd CDC*, December 9-12, Maui, USA.

Chen, W. and Saif, M. (2005). An Actuator Fault Isolation Strategy for Linear and Nonlinear Systems, *American Control Conference*, June 8-10, Portland, USA.

Chen, W. and Saif, M. (2006). Fault detection and isolation based on novel unknown input observer design, *American Control Conference*, June 14-16, Minneapolis, Minnesota, USA.

De Persis, C. and Isidori, A. (2001). A geometric approach to nonlinear fault detection and isolation, *IEEE Transactions on Automatic Control*, 46(6), pp. 853-865.

Edwards, C. and Spurgeon, S. (1994). On the development of a discontinuous observer, *International Journal of Control*, 59, pp. 1211-1229.

Fragkoulis, D., Roux, G. and Dahhou, B. (2007). Actuator fault isolation strategy to a waste water treatment process, *Conference on Systems and Control*, Mai 16-18, Marrakech, Morocco.

Fragkoulis, D., Roux, G. and Dahhou, B. (2008). Sensor fault detection and isolation scheme for single and multiple faults, *Internal Report LAAS N°08317*.

Gertler, J. J. (1998). *Fault Detection and Diagnosis in Engineering Systems*, Marcel Dekker Inc..

Frank, P. M. (1994). Online fault detection in uncertain nonlinear systems using diagnostic observers: a survey, *International Journal of System Science*, 12, pp. 2129-2154.

Hammouri, H., Kinnaert, M. and EI Yaagoubi, E. H. (1999). Observer based approach to fault detection and isolation for nonlinear systems, *IEEE Transactions on Automatic Control*, 44(10), pp. 1879-1884.

Isermann, R. (1994). On the applicability of Model-based fault detection for technical processes, *Control Engineering Practice*, 2, pp. 439-450.

Nejjari, F. (2001). Benchmark of an Activated Sludge Plant, *Internal report*, Terrassa, Spain.

Venkatasubramanian, V., Rengaswamy, R., Yin, K. and Kavuri, S. N. (2003). A review of process fault detection and diagnosis Part I: Quantitative model-based methods, *Computers and Chemical Engineering*, 27, pp. 293-311.

Xing-Gang, Y. and Edwards, C. (2005). Robust sliding mode observer-based actuator fault detection and isolation for a class of nonlinear systems, *Conference on Decision and Control*, and the *European Control Conference*, Seville, Spain, 12-15 December .

Batch Process Monitoring and Fault Diagnosis Based on Multi-Time-Scale Dynamic PCA Models

Yuan Yao* and Furong Gao**

** Dept. of Chemical and Biomolecular Engineering, Hong Kong University of Science and Technology, Clear water bay, Kowloon, Hong Kong SAR, P. R. China,*
*** Dept. of Chemical and Biomolecular Engineering, Hong Kong University of Science and Technology, Clear water bay, Kowloon, Hong Kong SAR, P. R. China*
(Tel: +852-2358-7139, Fax:+852-2358-0054, e-mail: kefgao@ust.hk).

Abstract: Dynamics are inherent characteristics of batch processes, which can be divided into short time-scale dynamics within a batch duration and long time-scale dynamics across several batches. The interactions between process variables make different types of dynamics confounded. Under such situations, it is difficult to perform efficient fault diagnosis. In this paper, a batch process monitoring scheme is proposed to separate different types of process variations for modeling and perform monitoring and fault diagnosis with multi-time-scale dynamic principal component analysis (PCA) models. Simulation results show that the fault diagnosis efficiency is enhanced.

Keywords: batch process, monitoring, fault diagnosis, principal component analysis, dynamics.

1. INTRODUCTION

In today's industrial manufacturing, batch processes are widely applied to manufacture high-value-added products. To ensure operation safety and product quality, the multivariate statistical monitoring methods, such as multiway principal component analysis (MPCA) (Nomikos and MacGregor, 1994; Nomikos and MacGregor, 1995) which is an extension of principal component analysis (PCA), have been utilized in batch process monitoring and fault diagnosis.

Dynamics are inherent characteristics of batch processes, including short time-scale dynamics within a batch duration and long time-scale dynamics across several batches. Different types of batch dynamics are usually caused by different values of variable response time which measures the time process variables take to react to given inputs. Fast-response variables have small response time constant, while slow-response variables have large values which may be longer than a batch duration. To model batch process dynamics better, several multivariate statistical monitoring methods have been proposed. Batch dynamic principal component analysis (BDPCA) (Chen and Liu, 2002) captures within-batch dynamic information, while two-dimensional dynamic principal component analysis (2-D-DPCA) (Lu, et al., 2005) can model both long and short time-scale dynamics in a two-dimensional (2-D) model structure.

In batch processes, variable correlations always exist. Especially, changes in slow-response variables can also affect fast-response variable trajectories. This makes different types of variable dynamics confounded, and causes difficulties in process fault diagnosis, as shown later. Therefore, it is desirable to have a method which can decouple process variation information according to dynamic time scales and monitor different types of variations separately. Thus, the fault diagnosis efficiency and accuracy can be enhanced.

Several existing multivariate statistical methods can divide process variations into blocks, scales or levels, but none of them can be utilized directly to handle the situation mentioned above. Multiblock PCA or partial least squares (PLS) methods (Westerhuis et al., 1998) group process variables into meaningful blocks and concern both the inner relationship within each block and the inter relationship among blocks. Although the variables with different response time can be divided into different blocks, two kinds of dynamics information are not separated due to variable correlations. Multiscale PCA (Bakshi, 1998) makes use of wavelet analysis techniques to transform each variable signal from time domain to frequency domain, and performs PCA on wavelet coefficients at each scale. However, the different dynamics characteristics of each variable are not taken into consideration. Multilevel component analysis (MLCA) and multilevel simultaneous component analysis (MLSCA) (Timmerman, 2006) separate within-batch variations and between-batch variations. But only the static variations are extracted, while process dynamics are not modeled. Besides, none of the methods reviewed in this paragraph can deal with long time-scale dynamics across several batches.

In this paper, a batch process monitoring scheme is developed. This scheme makes use of variable response time information which can be easily achieved, and separate process variations into different levels corresponding to dynamics time scales. 2-D-DPCA method is adopted to build multi-time-scale models. Thus, faults occurring to a certain level can be accordingly detected with the level model. Then, diagnosis can also be performed in the corresponding level, indicating the causing of the fault more clearly.

The article is organized as following. In section 2, the 2-D-DPCA method is reviewed. Then, a multi-time-scale batch process monitoring scheme is proposed and described detail in section 3. Simulation results are given in section 4. A batch process with both long time-scale and short time-scale dynamics is simulated to compare the monitoring and diagnosis efficiencies between the conventional 2-D-DPCA method and the proposed scheme. Finally, a conclusion is given in section 5 to summarize the paper.

2. TWO-DIMENSIONAL DYNAMIC PCA (2-D-DPCA)

2-D-DPCA method proposed by the authors can model both long and short time-scale batch process dynamics with a parsimonious two-dimensional (2-D) time series model structure together with PCA technique (Lu, et al., 2005).

Process dynamics can be indicated by the correlations between current measurements and lagged measurements. Long time-scale dynamics often behave as a kind of two-dimensional (2-D) dynamics, which means the current measurements are dependent not only on lagged measurements in the past time direction in the same batch, but also on lagged measurements in some past batches. These lagged variables form a region called the support region or the region of support (ROS). In 2-D-DPCA, an expanded data matrix $\tilde{\mathbf{X}}$ is formed by including all the lagged measurements in ROS, together with current measurements. For more details about ROS determination, please refer to Yao et al.'s work (2008).

Suppose $\tilde{\mathbf{X}}$ has been normalized to have unit variances and zero means. PCA algorithm is performed on it:

$$\tilde{\mathbf{X}} = TP^T + E \,. \tag{1}$$

where T and P are score matrix and loading matrix respectively, and E is the residual matrix. The number of scores retained in the score space can be determined using cross-validation (Wold, 1978). Thus, the original process data are divided into two subspaces. Score space extracts systematic variation information, including both 2-D dynamics and cross-correlation information among variables, while normal distributed noises are retained in residual space. Therefore, *SPE* statistic and corresponding control limits can be calculated for process monitoring in residual space. After a fault is detected by the *SPE* control plot, contribution plots with control limits (Westerhuis et al., 2000) are used in fault diagnosis to find the causes of the faults.

When a batch process only has short time-scale dynamics, its ROS is selected as a region containing several steps of lagged measurements in current batch. In such a case, 2-D-DPCA model is similar to BDPCA model (Chen and Liu, 2002).

3. MULTI-TIME-SCALE MONITORING SCHEME

3.1 Motivations

As mentioned in introduction section, in batch processes, fast-response variable trajectories are often affected by disturbances in slow-response variables. Take injection molding process as an example. In that process, temperature variables' response time constants are often longer than a batch duration, while pressure variables response fast. Suppose a disturbance occurs to barrel temperature. It takes a long time for barrel temperature to recover. During this period, the material properties, such as viscosity and density, change gradually due to the temperature change. This further causes slow drifts in pressure variable trajectories, although pressures are fast-response variables. From this example, it can be seen that both short and long time-scale dynamics are confounded in fast-response variable trajectories.

As shown in the simulation example in section 4, such confounding leads to difficulties in fault diagnosis results. Therefore, it is desirable to decouple process variation information into several levels according to dynamic time scales. Then, level models can be built and different types of variations can be monitored and diagnosed separately, so that the fault diagnosis efficiency and accuracy can be enhanced.

3.2 Variable classification

As a kind of external information, variable response time is easy to be estimated from process open-loop tests which are regular steps in controller designs. Such information is used to classify variables into groups. It is the first step of multi-time-scale modeling and monitoring.

In many cases, the variables can be simply divided into two groups. One contains fast-response variables, while the other contains slow-response variables which can cause long time-scale dynamics beyond a batch. In some other situations, it may be desired to further divide the above two groups into sub-groups. Suppose there are M number of variable divided into the fast-response variable group. Take each variable's response time constant as a pattern. The k-means clustering algorithm (Jain et al., 1999) is adopted for partitioning the M number of patterns. The final cluster number is determined automatically with a specified threshold of the minimal distance between two cluster centers or the maximal radius of a cluster. A larger threshold results in fewer variable groups; vice versa. The slow-response variable group can also be further divided in the same way. By doing so, the process variables with similar response time constants are clustered into the same group.

3.3 Multi-time-scale level separation

Without losing generality, first, suppose the process variables are divided into two groups. As discussed in section 3.1, two types of dynamics may confound in the trajectories of the variable in the fast-response variable group. To solve this problem, the operation data in this group should be decomposed into two parts: one part can be explained by the variable measurements in the slow-response variable group, and the other part can not be explained by them and only contains short time-scale dynamics. The level separation is based on the idea of external analysis, which was originally proposed by Takane and Shibayama (1991) and further

discussed by Yoon and MacGregor (2001). Kano et al. (2004) made use of this idea to distinguish faults from normal changes in operating conditions.

Consider a batch process data matrix $\hat{X}(I \times J \times K)$, where I, J, K are the number of batches, variables and time intervals respectively. Unfold this three-way data matrix into a two-way matrix $X(IK \times J)$ by keeping the variable dimension and merging the other two dimensions. Suppose X have been normalized. After variable classification, X can be described as $X = [F \ \ S]$, where F consists of J_F number of fast-response variables and S consists of $J_S = J - J_F$ number of slow-response variables. To decompose F, regression analysis is performed by regarding S and F as inputs and outputs respectively. If variables in S are independent of each other, the ordinary least square (OLS) regression can be used:

$$\Phi = (S^T S)^{-1} S^T F, \tag{2}$$

where Φ is the regression coefficient matrix. The significance of regression can be tested (Montgomery, 2005) to show whether there are correlations between S and F. If there is no correlation, the levels are naturally seperated. The short time-scale level consists of $D^S = F$, while the long time-scale level consists of $D^L = S$. Otherwise, calculate (3).

$$E = F - S\Phi, \tag{3}$$

where $S\Phi$ contains a part of information in F which is explained by slow-response variable, while the filtered data matrix E dose not contains long time-scale dynamics. When the slow-response variables are not independent, PLS or principal component regression (PCR) can be utilized to avoid the collinearity problem. Thus, the process variation information is separated into two levels according to different time scales of dynamics: $D^S = E$ and $D^L = [S\Phi \ \ S]$.

When there are more than two groups, the time-scale level separation is performed in an iterative way. Unfolded data matrix X is described as $X = \left[X_1^0 \quad X_2^0 \quad \cdots \quad X_C^0 \right]$, where X_i^j is the filtered data matrix of the ith variable group after the jth iteration run in time-scale level separation, consisting of J_i number of variables. When $j = 0$, X_i^j represents the data before performing iteration steps. C is the total number of variable groups, and the variables in X_i^j response faster than the variables in X_{i+1}^j. In the jth run, let $S^j = X_{C-j+1}^{j-1}$ and $F^j = \left[X_1^{j-1} \quad X_2^{j-1} \quad \cdots \quad X_{C-j}^{j-1} \right]$. Φ^j is then calculated in the similar way as (2), and the data are filtered as

$$\begin{aligned} E^j &= F^j - S^j\Phi^j = \left[X_1^{j-1} \quad X_2^{j-1} \quad \cdots \quad X_{C-j}^{j-1} \right] - X_{C-j+1}^{j-1}\Phi^j \\ &= \left[X_1^j \quad X_2^j \quad \cdots \quad X_{C-j}^j \right] \end{aligned}. \tag{4}$$

After C-1 cycles of iteration, all levels are separated. The shortest time-scale level consists of $D^1 = E^{C-1}$. The second shortest time-scale level consists of $D^2 = [S^{C-1}\Phi^{C-1} \ \ S^{C-1}]$. … The longest time-scale level consists of $D^C = [S^1\Phi^1 \ \ S^1]$.

3.4 Multi-time-scale dynamic PCA modeling, monitoring and fault diagnosis

After level separation, 2-D-DPCA is adopted to construct level models for online monitoring and fault diagnosis.

Take a C level separation as an example. In level j (j>1), $D^j = [S^{C-j+1}\Phi^{C-j+1} \ \ S^{C-j+1}]$. Since $S^{C-j+1}\Phi^{C-j+1}$ is completely dependent on S^{C-j+1}, it only represents redundant information in a process monitoring context. Therefore, the variation information in each level is reorganized as $G^1 = E^{C-1}$, $G^2 = S^{C-1}$, …, $G^C = S^1$ with matrix dimensions of $(IK \times J_1)$, $(IK \times J_2)$, …, $(IK \times J_C)$ respectively. These matrices are rearranged into three-dimensional arrays with dimensions of $(I \times J_1 \times K)$, $(I \times J_2 \times K)$, …, $(I \times J_C \times K)$. Then, following ordinary procedures, 2-D-DPCA models can be established for each level. The *SPE* control limits are calculated for online monitoring. For a level belonging to short time-scale dynamics, the 2-D-DPCA model reduces to a BDPCA model. For these levels, the T^2 control limits can also be calculated, since there is no batch-wise dynamics.

In online monitoring, the new data are firstly filtered based on (4) using coefficient matrices Φ^1, Φ^2, …, Φ^{C-1} in turns. Thus, the variations contained in the new data are separated into different time-scale levels. The corresponding 2-D-DPCA model is utilized to monitor each level. After faults are detected in some levels, the contribution plots can be used for fault diagnosis in these levels accordingly.

4. SIMULATION EXAMPLE

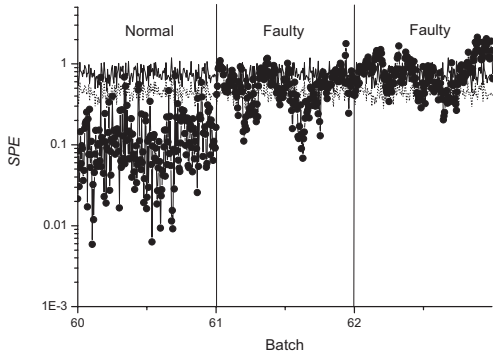
4.1 Batch process modeling

In this section, a simulated batch process with both long and short time-scale dynamics is utilized to compare the monitoring and fault diagnosis efficiency of the proposed multi-time-scale dynamic PCA models with the conventional 2-D-DPCA model. The process model is given as below,

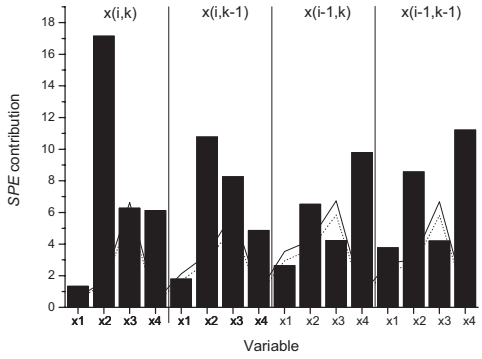
$$\begin{aligned} x_1(i,k) &= 0.5 * x_1(i,k-1) + 0.8 * x_1(i-1,k) - 0.3 * x_1(i-1,k-1) \\ x_2(i,k) &= 0.44 * x_2(i-1,k) + 0.67 * x_2(i,k-1) - 0.11 * x_2(i-1,k-1) \\ x_3(i,k) &= 0.4 * x_3(i,k-1) + 0.25 * x_1(i,k) + 0.35 * x_2(i,k) \\ x_4(i,k) &= 0.8 * x_4(i,k-1) + 0.53 * x_1(i,k) - 0.33 * x_2(i,k) \end{aligned}, \tag{5}$$

where i is the batch index; k is the time index; x_1 and x_2 are two independent slow-response variables with long time-scale dynamics described in a 2-D structure; x_3 and x_4 are fast-response variables correlated to their own values at one step before in the current batch, which are also affected by x_1, x_2. Gaussian noises with variance 0.01 are added into the data.

For conventional 2-D-DPCA modeling, the ROS is determined as $\mathbf{x}(i,k-1), \mathbf{x}(i-1,k), \mathbf{x}(i-1,k-1)$, where $\mathbf{x}(i,k-1) = \left[x_1(i,k) \quad x_2(i,k) \quad x_3(i,k) \quad x_4(i,k) \right]$. So that, there are totally 16 variables in the augmented data matrix $\tilde{\mathbf{X}}$, including 4 current variables and 12 lagged variables in the ROS.

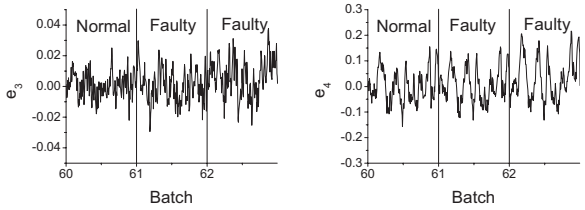


(a)



(b)

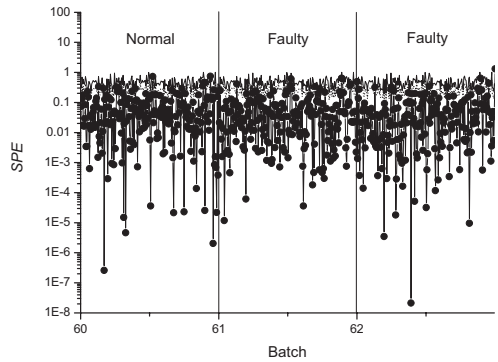
Fig. 1. Monitoring and diagnosis results of fault 1 based on 2-D-DPCA: (a) monitoring result; (b) fault diagnosis result.



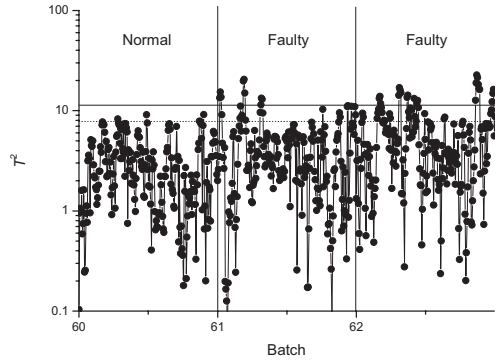
(a) (b)

Fig. 2. Filtered variable trajectories in fault 1: (a) e_3; (b) e_4.

For multi-time-scale dynamic PCA modeling, x_1 and x_2 belong to the slow-response variable group S, while x_3 and x_4 are divided into the fast-response variable group F. The regression model between F and S is built to remove the effects of x_1 and x_2 from x_3 and x_4, as described in (2) and (3). Supposing e_3 and e_4 are the filtered values of x_3 and x_4, the variation information is separated into $G^S = \begin{bmatrix} e_3 & e_4 \end{bmatrix}$ as the short time-scale level and $G^L = \begin{bmatrix} x_1 & x_2 \end{bmatrix}$ as the long time-scale level. Then, 2-D-DPCA is performed on each level to model the two different types of dynamics. Let $\hat{\mathbf{x}}(i,k-1) = \begin{bmatrix} x_1(i,k) & x_2(i,k) \end{bmatrix}$. In the long time-scale level, the ROS is selected as $\hat{\mathbf{x}}(i,k-1), \hat{\mathbf{x}}(i-1,k), \hat{\mathbf{x}}(i-1,k-1)$.



(a)



(b)

Fig. 3. Monitoring results of fault 1 based on short time-scale level model: (a) *SPE* plot; (b) T^2 plot.

The 2-D-DPCA model is calculated based on 2 current variables in $\hat{\mathbf{x}}(i,k)$ and 6 lagged variables in the ROS. In the short time-scale level, the algorithm is performed on 4 variables including $e_3(i,k), e_4(i,k), e_3(i,k-1), e_4(i,k-1)$.

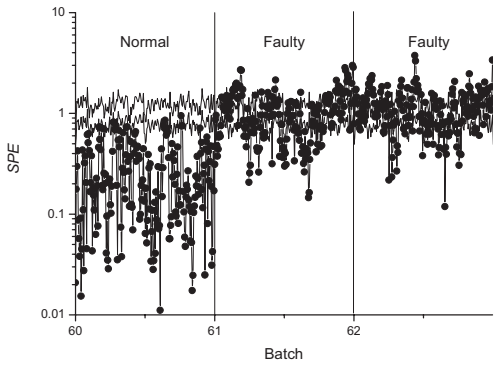
4.2 Online modeling and fault diagnosis

Two faults are introduced into the process. Fault 1 occurs to the slow-response variable x_2. From batch 61, x_2 is formulated as (6) to simulate a fault:

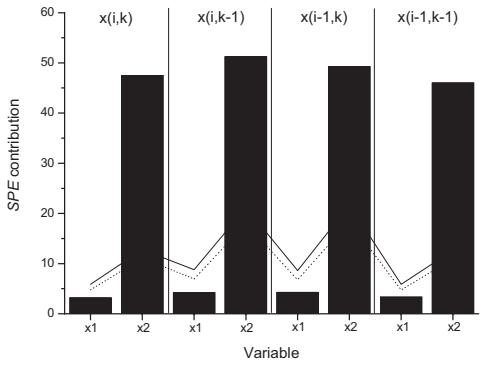
$$x_2(i,k) = 0.6 * x_2(i-1,k) + 0.3 * x_2(i,k-1) + 0.2 * x_2(i-1,k-1) . \quad (6)$$

Fig. 1 shows the monitoring and the fault diagnosis results based on conventional 2-D-DPCA, respectively. The *SPE* control chart shows that the fault can be detected from the beginning of batch 61. However, from the contribution plot of batch 61, Fig. 1(b), it is hard to say which variable is faulty. Due to the variable correlations, many variables (including the lagged variables) are outside the control limits.

In multi-time-scale monitoring, variable x_1 and x_2 are filtered to get short time-scale dynamic signals e_3 and e_4. Since the fault occurs to the slow-response variable x_2, and the effects



(a)



(b)

Fig. 4. Monitoring results of fault 1 based on long time-scale level model: (a) monitoring; (b) diagnosis.

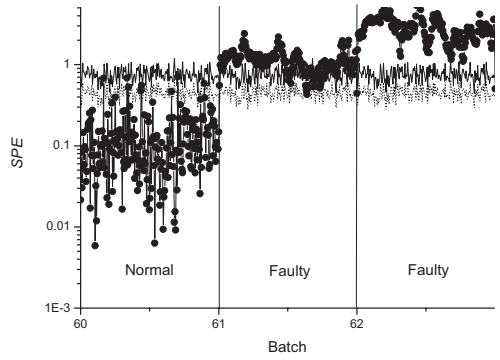
of x_1 and x_2 have been removed from the short time-scale level, there is no significant difference between the trajectories of e_3 and e_4 in a normal cycle and those in the faulty cycles, as shown in Fig. 2. The monitoring results in Fig. 3 confirm this. Neither *SPE* nor T^2 plot in this level is affected by the fault significantly. At the same time, the *SPE* control plot in the other level detects the fault efficiently, as Fig. 4(a) shows. This points out that the fault happens in the long time-scale level. Then, contribution plot in this level is plotted to find out the reason of the fault. From Fig. 4(b), it is very easy to conclude that x_2 is the faulty variable.

Fault 2 is about the fast-response variable x_3. From batch 61, the formulation of x_3 becomes:

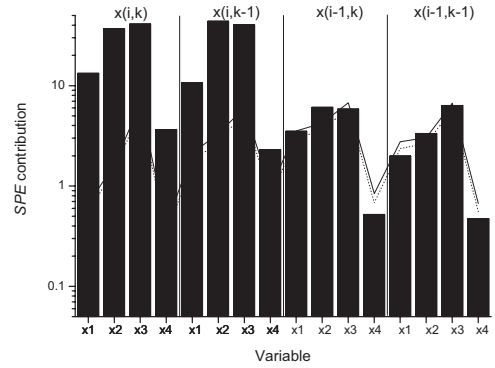
$$x_3(i,k) = 0.5 * x_3(i-1,k) + 0.25 * x_1(i,k) + 0.35 * x_2(i,k). \quad (7)$$

As shown in Fig. 5, again, the conventional 2-D-DPCA detects the fault very quickly, but the contribution plot can not give a clear indication about the reason of the fault.

Fig. 6 shows the trajectories of e_3 and e_4. Obviously, significant magnitude differences exist between the trajectory of e_3 in a normal batch and that in faulty batches. So that, this fault is hopefully to be detected by the T^2 control chart in the short time-scale level, which is confirmed by Fig. 7(a). The

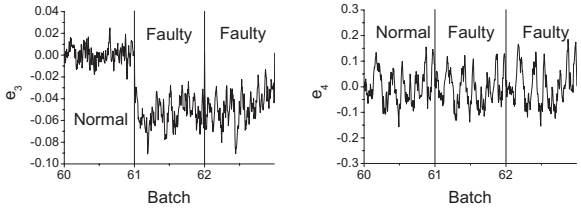


(a)



(b)

Fig. 5. Monitoring and diagnosis results of fault 2 based on 2-D-DPCA: (a) monitoring; (b) diagnosis.



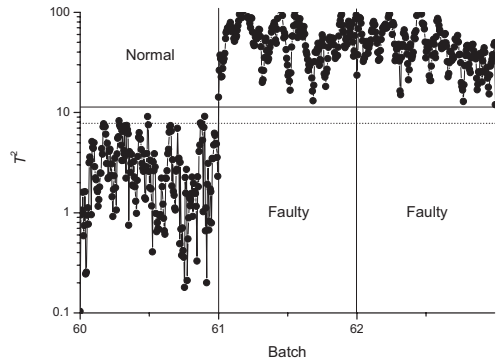
(a) (b)

Fig. 6. Filtered variable trajectories in fault 2: (a) e_3; (b) e_4.

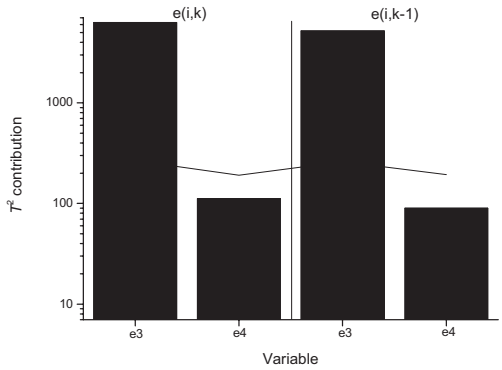
monitoring in the other level, as shown in Fig. 8, dose not show the fault, as it only occurs to a fast-response variable and dose not affect the long time-scale dynamics. The fault diagnosis is only needed to be performed in the short time-scale level. The contribution plot diagnoses the reason of the fault clearly and correctly, as Fig. 7(b) shows.

5. CONCLUSIONS

Batch process variables have various response time constants, causing dynamics with different time scales. The trajectories of the fast-response variables are often affected by the slow-



(a)



(b)

Fig. 7. Monitoring and diagnosis results of fault 2 based on short time-scale level model: (a) monitoring; (b) diagnosis.

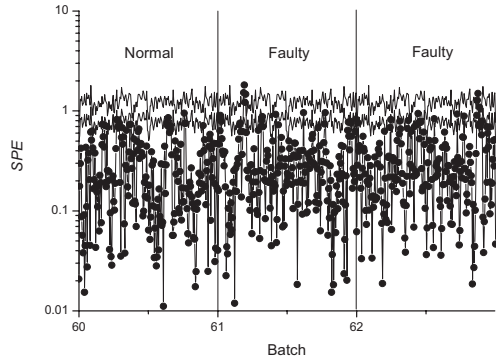


Fig. 8. Monitoring results of fault 2 based on long time-scale level model

response variables, confounding different types of dynamics and causing trouble in fault diagnosis.

A multi-time-scale dynamic PCA monitoring scheme is proposed in this paper. The process variations are separated into different levels according to the dynamics time scales. Then 2-D-DPCA method is adopted to model each level for online monitoring. The simulation results show that the fault diagnosis accuracy is largely improved.

In this paper, variable response time constants are assumed to be known as a kind of external information. It is better if such information can be achieved from the analysis of the operation data. This issue will be studied in the future researches to make the method completely data-based.

REFERENCES

Bakshi, B.R. (1998). Multiscale PCA with application to multivariate statistical process monitoring. *AIChE Journal*, 44, 1596-1610.

Chen, J. and Liu K.C. (2002). On-line batch process monitoring using dynamic PCA and dynamic PLS models. *Chemical Engineering Science*, 57, 63-75.

Jain, A.K., Murty, M.N., and Flynn, R.J. (1999). Data clustering: a review. *ACM Computing Surveys*, 31, 264-323.

Kano, M., Hasebe, S., Hashimoto, I. and Ohno, H. (2004). Evolution of multivariate statistical process control: application of independent component analysis and external analysis. *Computers and Chemical Engineering*, 28, 1157-1166.

Lu, N., Yao Y., and Gao F. (2005). Two-dimensional dynamic PCA for batch process monitoring. *AIChE Journal*, 51, 3300-3304.

Montgomery, D.C. (2005). *Design and analysis of experiments*, *6th ed*, New York: Wiley.

Nomikos, P. and MacCregor J.F. (1994). Monitoring batch processes using multiway principal component analysis. *AIChE Journal,* 40, 1361-1375.

Nomikos, P. and MacCregor J.F. (1995). Multivariate SPC charts for monitoring batch processes. *Technometrics,* 37**,** 41-59.

Takane, Y. and Shibayama, T. (1991). Principal component analysis with external information on both subjects and variables. *Psychometrika*, 56, 97–120.

Timmerman, M.E. (2006). Multilevel component analysis. *British Journal of Mathematical and Statistical Psychology*, 59, 301-320.

Westerhuis, J.A., T. Kourti, and J.F. MacGregor (1998). Analysis of multiblock and hierarchical PCA and PLS models. *Journal of Chemomet*rics, 12, 301-321.

Westerhuis, J.A., S.P. Gurden and A.K. Smilde (2000). Generalized contribution plots in multivariate statistical process monitoring. *Chemometrics and Intelligent Laboratory Systems*, 51, 95-114.

Wold, S. (1978). Cross-validatory estimation of the number of components in factor and principal components models. *Technometrics*, 20, 397-405.

Yao, Y., Diao, Y., Lu, N., Lu, J., and Gao, F. (2008). Two-dimensional dynamic principal component analysis wieht auto-determined suppert region. *Industrial & Engineering Chemistry Research*, in print.

Yoon, S. and MacGregor, J.F. (2001). Incorporation of external information into multivariate PCA/PLS models. *Proc. of 4th IFAC Workshop on On-line Fault Detection and Supervision in the Chemical Industries*, 121-126.

FAULT DECTION AND VARIATION SOURCE IDENTIFICATION BASED ON STATISTICAL MULTIVARIATE ANALYSIS

Ming-Da Ma*. Chun-Cheng Chang. Shi-Shang Jang**. David Shan-Hill Wong****
Sheng-Tsaing Tseng***

**Center for Control and Guidance Technology, Harbin Institute of Technology*
Harbin, China, (e-mail: mamingda@hit.edu.cn).
*** Department of Chemical Engineering, National Tsing-Hua University, Hsin-Chu, Taiwan*
(e-mail: ssjang@mx.nthu.edu.tw; dshwong@che.nthu.edu.tw)
**** Institute of Statistics, National Tsing-Hua University, Hsin-Chu, Taiwan,*
(e-mail: sttseng@stat.nthu.edu.tw)

Abstract: This paper aims to solve the problems of fault diagnosis and variation reduction by using multivariate statistical techniques when the quality measurements are scarce. Both single stage process and multi-stage process are considered. For the single stage process, the nonparametric statistical method, Wilcoxon rank-sum test is used to identify the key variable/step that causes the fault of the un-qualified wafers. For the multi-stage process, the most important variables are first picked out by systematic statistical analysis, and the specifications of these key variables are designated using nonparametric method to improve the product yield. Gene map which gives visual images is used to assist the analysis. Industrial examples are given to show the effectiveness of the proposed method.

Keywords: semiconductor manufacturing, fault detection, Wilcoxon rank-sum test, cluster analysis, stepwise regression.

1. INTRODUCTION

State-of-the-art semiconductor processes are often pushed to the limits of current technologies, resulting in processes that have little or no margin for error. Advanced process control (APC) and fault detection and classification (FDC) are widely applied in semiconductor industries to reduce cycle-time and improve yield. The focus of this paper is on the fault detection algorithms to find out the variation source and root-cause of scrap wafers by using statistical multivariate analysis techniques.

Detection of process and tool faults in the shortest possible time is critical for minimizing scrap wafers and improving product yields for semiconductor manufacturing. However, most of wafer-states lack in situ sensor to provide real time information and usually are measured offline and less frequently than every wafer, which can lead to a number of scrapped wafers before a fault is detected. In the meanwhile, fortunately, more and more real time measurements of manufacturing equipments like temperature, pressure, power and flow rate, etc., are available due to the advances in metrology technology. These real time measurements provide valuable information about the tool status and can be used to predict final wafer characteristics. Further, it also provides a way to improve product quality by detecting and identifying equipment malfunctions in real time without interrupting the normal operations. The difficulty is, with such an abundant amount of data available, it is usually not clear which tool-state variable is critical or closed related with the final product quality.

Principal component analysis (PCA) and partial least squares (PLS) have drawn increasing interest and have been studied extensively in semiconductor manufacturing industry. PCA and PLS are useful tools for data compression and information extraction and have the advantages of dealing with high dimension and collinearities. PCA/PLS methods find linear combinations of variables that describe major trends in a data set. Considering the batch nature of semiconductor manufacturing, multi-way PCA is usually used to unfold three dimensions data into 2-D data array (Macgregor, 1994). Yue et al. applied multi-way PCA method to optical emission spectra for plasma etchers.

Most of the methods mentioned above require a large amount of training data to build a reliable statistical model to capture the key characteristics of the process. However, in real practice, many fabs are operating with diversified products of small account (Ma, et al., 2008) which means that one has to find out the causes of un-qualified wafers with limited quality data. Compared with principal components which are combined by all process variables, engineers are more eager to know which variable exactly, or linear combinations of several variables, plays an important role on the product quality. It is also of interest to know which step is critical to the whole streamline.

In this paper, a systematic approach is proposed for fault diagnosis and variation reduction by using statistical multivariate techniques. Both single stage process and multi-

stage process are considered. For the single stage process, the nonparametric statistical method, Wilcoxon rank-sum test is used to identify the key variable/step that causes the fault of the un-qualified wafers. For the multi-stage process, homogeneous process variables are first grouped by using cluster analysis, and representative variables or linear combinations of variables of each group are picked out. Then the key clusters are selected by stepwise regression method. Further, the upper and lower limits of these selected representative variables are designated to reduce product variation. It is shown that the proposed method improves the product yield substantially.

Recently, combinatory and high throughput experiments have received widespread attention in biology. Synopsis of large amount of experiment data and subsequent information mining from such data has become a special branch of study known as bioinformatics (Baldi and Brunak, 2001). The key experimental technique that is responsible for the advancement of bioinformatics is the microarray which enables expressions of tens of thousands of genes be measured and represented on a small array of colored image dots. In this paper, we demonstrate that quick diagnosis of the key variable/step that causes the fault in final quality can be achieved by simple statistical analysis of measured values of different sensors and graphical synopsis of results of such analysis. Furthermore, specifications for the key variables, which are usually far from optimal in original settings, can be designated to improve the product yield.

2. FAULT DETECTION FOR SINGLE STAGE PROCESS

2.1 Problem statement

Consider quality data of n wafers are collected from a tool, n_1 wafers are qualified, and n_2 wafers are un-qualified, hence $n_1+n_2=n$. Let's denote that m steps with v variables are implemented during the whole process. It is assumed that in each step t_s seconds are carried out for some certain objective (for instance temperature ramped up, current ramped down,…,etc.), where s=1,…,m. Suppose that the total time for all steps is t, then $t_1 + t_2 + \cdots + t_m = t$; let $T_r = t_1 + \cdots + t_r$, where r=1,…,m. Now, let's define $X_{i,j,k,l}$ to be the kth independent variable at batch time l of jth wafer, where j=1,…,n_i, k=1,…,v, l=1,…,t, and i=1 means the wafer is qualified, i=2 indicates the wafer is not qualified. Now, the problem is what is the p-value of $X_{i,j,k,l}$ to distinguish the wafer is qualified or unqualified in case n_1 and n_2 are small.

2.2 Statistical analysis

It is general to apply t-test to distinguish two set of data whether or not their mean is equal to each other. However, in this case n_1 and n_2 are small, a two sample t-test is not appropriate since the above two set of data may not be in normal distribution. Therefore, a nonparametric analysis, Wilcoxon rank-sum test, is used here.

The Wilcoxon rank-sum test is a nonparametric alternative to the two-sample t-test which is based solely on the order in which the observations from the two samples fall (Higgins, 2004). It is valid for data from any distribution, whether normal or not, and is much less sensitive to outliers than the two-sample t-test. The Wilcoxon test is based upon ranking the n_1+n_2 observations of the combined sample. Each observation has a rank: the smallest has rank 1, the 2nd smallest rank 2, and so on. The Wilcoxon rank-sum test statistic is the sum of the ranks for observations from one of the samples.

In this work, we implement Wilcoxon rank-sum test to find the p-value of the hypothesis of

$$H_0 : \mu_{1,k,l} = \mu_{2,k,l} \quad vs. \quad H_a : \mu_{1,k,l} \neq \mu_{2,k,l} \qquad (1)$$

where $\mu_{1,k,l}$ and $\mu_{2,k,l}$ are the mean of qualified and un-qualified wafers of the kth variable at time l respectively. It is assumed that there is not much prior knowledge of the product and no evidence shows that $\mu_{1,k,l}$ is greater or smaller than $\mu_{2,k,l}$. Therefore, a two-side test is implemented here.

Let $p_{k,l}$ be the above p-value of the kth variable at time l, three different approaches to evaluate the above approach can be implemented

(i) Evaluate the p-value of a process variable by finding the average p-value of the process variable in the whole time horizon:

$$P_k = \sum_{l=1}^{t} p_{k,l} \Big/ t \qquad (k=1,2,\ldots,v) \qquad (2)$$

(ii) Evaluate the average of p-value of each variable at each step

$$P_{k,b} = \sum_{l=T_{b-1}+1}^{T_b} P_{k,l} \Big/ t_b \qquad (k=1,2,\ldots,v\ ;b=1,2,\ldots.m,) \qquad (3)$$

(iii) Direct observe the p-value $p_{k,l}$ of each variable at each different time.

From the statistical analysis of the above approaches, we can determine which process variable, which step, plays an important role on the quality of the wafers, and more specifically, which second is critical for the final product quality. All these information is valuable to the engineers for their further improvement of the product quality.

2.3 Illustrative example

The proposed algorithm is applied to a high-density plasma chemical vapor deposition (HDP-CVD) process. HDP-CVD, which is used as the gap-filling process for the dielectric in semiconductor circuits, features a high gap-fill capability compared with conventional plasma CVD by the excitation

of a high-density plasma. The schematic diagram of the HDP-CVD reactor is shown in Fig. 1.

There are 33 process variables for this manufacturing process. 9 steps are implemented for this process and the processing time is shown in Table 1. The quality data obtained from WAT test of 25 wafers are collected, among which 21 wafers are qualified and 4 wafers are un-qualified.

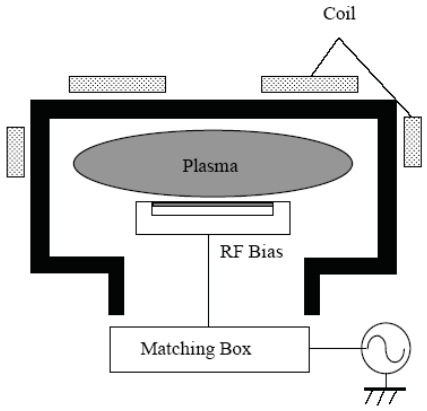


Fig. 1. Schematic diagram of the HDP-CVD reactor.

Table 1 Processing time of each step

Step	1	2	3	4	5	6	7	8	9	Total
Seconds	5	30	53	3	67	5	10	15	5	193

The proposed statistical method is applied to this process. The p-value of hypothesis (1) is calculated for process variables. To find out the key process variable that plays an important role on the process, equation (2) is implemented and the image plot of 1-P_k is shown in Fig. 2. From Fig. 2, we can determine which process variable is more influential for the product quality. This industrial gene map can help engineers to determine which process variable is important and which one is less important at a first glance. Engineers can grasp as much as information in the shortest time with the help of industrial gene map.

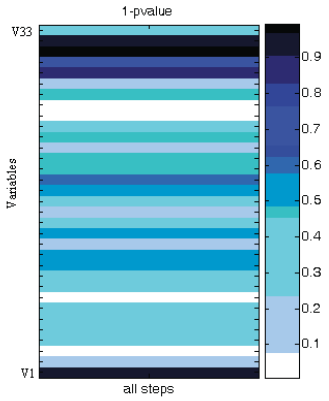


Fig. 2. Image of 1-P_k of total average approach

To further know which step is critical for the process, equation (3) is evaluated for the profile variables of the process and the result is shown in Fig. 3. Similarly, Fig. 3 corresponds to a matrix of dimension 33 by 9. Obviously, the industrial gene map is more visual and straightforward. From Fig. 3, it is observed that most of critical steps are also related with the settings of temperature. The p-value of the profile variables in second are shown in Fig. 4. It can help engineers know when a fault is most likely to happen.

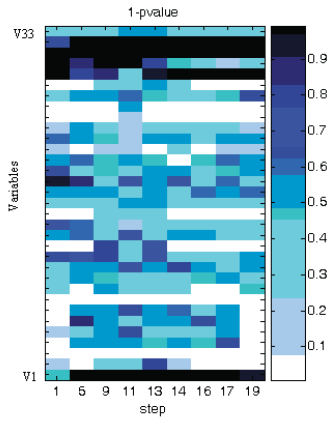


Fig. 3. Image plot of 1-$P_{k,b}$ of step average approach

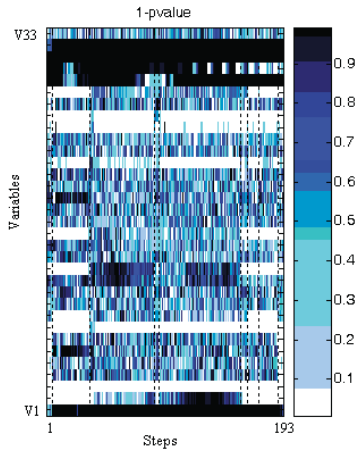


Fig. 4. Image plot of 1-$p_{k,l}$

3. VARIATION REDUCTION FOR MULTI-STAGE PROCESS

3.1 Problem statement

In this section, a statistical method is proposed to find out the key variables that have essential effects on the product quality for the multi-stage manufacturing process. Similarly, the basic assumption is that there is relatively few quality data available compared with process variables. Then, specifications for the key variables which are usually far from optimal in original settings are designated to improve the

product yield. This framework provides a systematic method of drawing inferences from the available evidence without interrupting the normal process operation. The proposed method is directly illustrated by an industrial example. The statistical methods used in the following analysis include cluster analysis, canonical correlation analysis and stepwise regression.

3.2 Statistical analysis and illustrative example

Consider a CVD process. Every wafer must be processed by three chambers A, B, and C successively. Denote the process variables of chamber A, B and C as X_A, X_B and X_C, respectively. The final quality variable is denoted as Y which may contain wafer thickness measurements and wafer electrical measurements. In the following analysis, the method is illustrated for the wafer thickness y, which is one of the most important characteristics of wafers.

The numbers of steps and process variables for chamber A, B and C are list in Table 2. The data set includes measurements of 526 wafers from 22 batches. In this analysis, we want to know which variable, of which chamber, on which step, has an essential effect on the wafer thickness. Every process variable from different chambers on different is treated as an independent variable. Therefore, it is still the case that there are much more process variables than the quality data. Furthermore, process variables are usually highly correlated because of physical and chemical principles governing the process operation. To pick out the most influential variables for the quality variable y, the first step is to reduce the redundancy of the original data set.

Table 2 Number of steps and variables of the three chambers

Chambers	Number of steps	Number of variables
A	13	79
B	5	19
C	12	79

Cluster analysis is a useful technique used for combining observations into groups or clusters such that each group or cluster is homogeneous with respect to certain characteristics. Simultaneously, each group should be different from other groups with respect to the same characteristics (Sharma, 1996). The definition of similarity or homogeneity varies from analysis to analysis, and depends on the objectives of the study. In this study, it is desired to combine variables that are highly correlated into one group. Therefore, the similarity measure is defined as

$$d_{ij} = 1 - \left| r_{ij} \right| \qquad (4)$$

where r_{ij} is the correlation coefficient of variables x_i and x_j. For variables that are highly correlated, d_{ij} would be small which represents similarity and vice versa. The clustering method adopted here is average-linkage method, one of the hierarchical clustering methods.

To determine the number of clusters, the rule that the correlation coefficient of the variables from the same groups should be greater than 0.9 is used. The result of cluster analysis is shown in Fig. 5-7. In these figures, variables that are filled with the same color or indicated with the same number are of the same group.

Then, the next step is to select representative variables from each group. The variables picked out should give good variance explanation which is usually evaluated by the R^2 statistics of the wafer thickness y. For example, the R^2 statistics of one variable selected from group 8 is 0.352 and the total R^2 of the whole group is 0.361. In such case, one process variable is capable of representing the group.

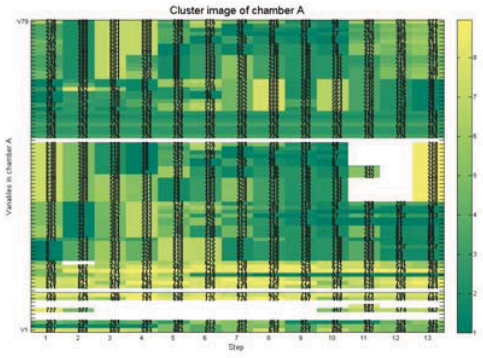


Fig. 5. Cluster image of chamber A.

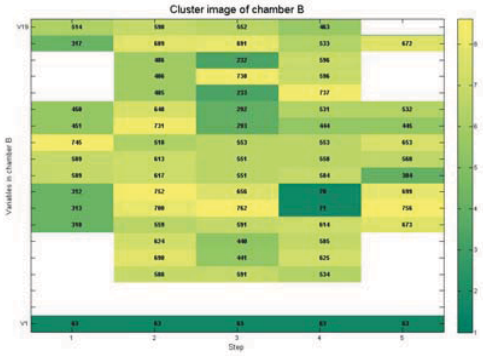


Fig. 6. Cluster image of chamber B.

However, in some circumstances, the R^2 of each individual variable is quite low yet the linear combination of these variables contributes a high R^2. In this case, it is more appropriate to use linear composites of the original variables to represent the group. This problem actually belongs to the field of canonical correlation analysis. The new variables, the linear composites, are called canonical variates. The coefficients of the canonical variates are determined to make the correlation between the linear composites maximum. For this special case, there is only one quality variable, the wafer thickness y. Therefore, canonical correlation analysis is essentially equal to the linear multiple regression.

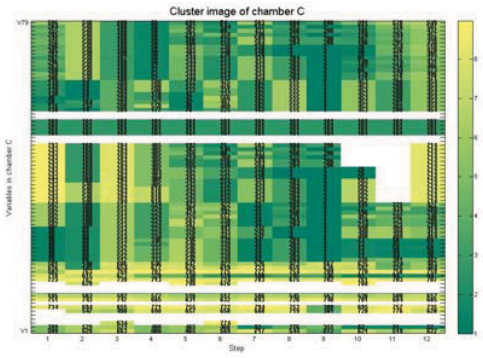


Fig. 7. Cluster image of chamber C.

Then, the question is, when a single variable should be used and when a linear composite should be used to represent a group. In this application, the following rules are adopted: if the R^2 of individual variable is more than eighty percent of the total R^2, then the single variable which has the largest R^2 is used to represent the whole group; otherwise, a linear composite is used. The number of variables in the canonical variate is increased till the R^2 of the linear composite is more than eighty percent of the total R^2. The coefficients of the linear composite are obtained from canonical correlation analysis.

After picking out the representative variable from each group, the next step is to select important representative variables from all the groups. The method used is stepwise regression. Stepwise regression is a statistical method used for variable selection in linear regression. The procedure iteratively constructs a sequence of regression models by adding or removing variables at each step. The criterion for adding or removing a variable at any step is usually expressed in terms of a partial *F*-test Montgomery, et al., 2001). The changes of R^2 and adjusted R^2 of stepwise regression are shown in Fig. 8. There are 68 representative variables selected by the stepwise regression.

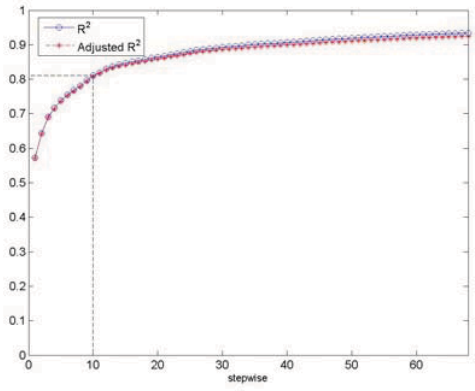


Fig. 8. Representative variables selected by stepwise regression method.

It is not an easy task to monitor 68 variables online simultaneously. Therefore, the first ten representative variables selected by stepwise regression are picked out and analyzed. The first ten representative variables listed in Table 8 give a good variance explanation because the R^2 and adjusted R^2 are higher than 0.8 which can be seen from Fig. 8.

In fact, all the 526 wafers are qualified wafers. To reduce the variance of wafer thickness further, we define $\left[\overline{y}-1.5s_y, \overline{y}+1.5s_y\right]$ as the acceptable region for the wafer thickness. Here, $\overline{y}$ is the average value of y and s_y is the standard deviation of y, respectively. The wafers fall out of this region is treated as "un-qualified" now. Among all the 526 wafers, there are 455 wafers fall into the acceptable region. Therefore, the yield is 0.865. In the following analysis, we will develop a nonparametric method to find out the new specifications for the above ten important representatives to improve the product yield.

First, the center point for all the qualified wafers in a space defined by the 10 important representative variables is determined. The Mahalanobis distance of each qualified wafer from the center point is calculated as

$$MD_i = \left(X_i - \mu\right)^T S \left(X_i - \mu\right) = c_i \quad (5)$$

where X is a 10×1 vector of coordinates and S is a 10×10 covariance matrix, μ is the center point. Then, the yield can be viewed as an implicit function of the Mahalanobis distance. Each value of Mahalanobis distance corresponds to a value of yield which is defined as the ratio between the number of qualified wafers and the number of all the wafers within the Mahalanobis distance. A graphical interpretation of this relationship is shown in Fig. 9. In this figure, the solid line is the relationship between the yield and the Mahalanobis distance and the dashed line is its 95% confidence interval.

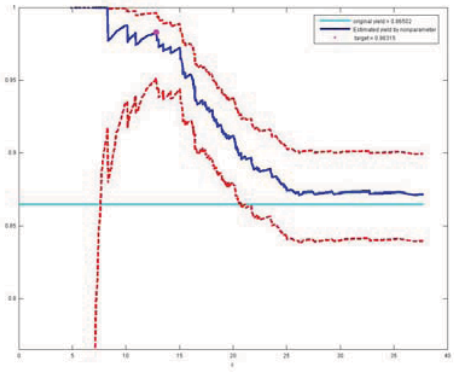


Fig. 9. Plot of Mahalanobis distance versus product yield.

It can be observed that the yield is not reliable when c_i is small because the samples within the corresponding Mahalanobis distance are few. To get a balance between reliability and high yield, the point corresponds to one third of the maximum of c_i which is marked as a dot in Fig. 9 is

used to derive the specifications of the ten representative variables. Once c_i is determined, the joint boundary of the ten representative variables is also determined.

However, the joint boundary which is a function of ten independent variables can not be easily monitored. Therefore, the projections of the joint boundary onto the axes of coordinates are used as the new specifications of the ten representative variables. The yield increased greatly when the upper and lower bounds of the first representative variable are designated. A graphical interpretation of the increase of the yield is shown in Fig. 10. The increases of the yield are not obvious after the designation of the specification of the third representative variable.

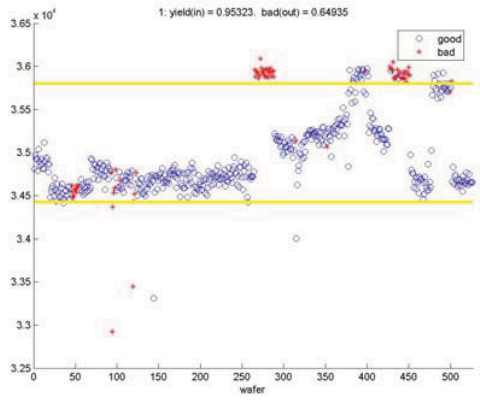


Fig. 10. Specifications of the first representative variable.

It is of interest to study the improvement of process capability ratio after the specifications of the ten representative variables are designated. The process capability ratio (PCR, or C_p) is defined as

$$C_p = \frac{USL - LSL}{6\sigma} \qquad (6)$$

where USL and LSL are the upper and lower specification limits, respectively. Since σ is unknown, it is replaced by the standard deviation s. If the process capability ratio and standard deviation are treated as a function of Mahalanobis distance, then we can get

$$\frac{C_p(c_i)}{C_p} = \frac{s}{s(c_i)} \qquad (7)$$

The relationship between $C_p(c_i)/C_p$ and the Mahalanobis distance is shown in Fig. 11. It can be observed that there is about 40% improvement of process capability ratio for the point we used to designate the specifications of the representative variables. The changing trend of $C_p(c_i)/C_p$ is consistent in the area where the point we used also indicates that the value of c_i we chose is appropriate.

4. CONCLUSIONS

Nowadays, many semiconductor manufacturing foundries are operating with diversified products of small account which makes the fault detection and variation reduction difficult. In this paper, systematic statistical methods are proposed to solve this difficulty. Both single stage process and multi-stage process are considered. The effectiveness of the proposed methods are illustrated by industrial examples.

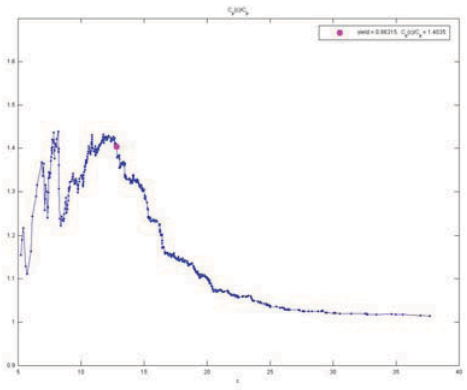


Fig. 11. Plot of Mahalanobis distance versus $C_p(c_i)/C_p$.

REFERENCES

Cherry, G.A. and Qin, S.J. (2006). Multiblock principal component analysis based on a combined index for semiconductor fault detection and diagnosis. *IEEE Trans. Semiconduct. Mnauf.*, 19 (2), pp 159-172.

Ma, M.D., Chang, C.C., Wong, D.S.H., and Jang, S.S. (2008). Identification of tool and product effects in a mixed product and parallel tool environment. *J. Process Contr.*, doi:10.1016/j.jprocont.2008.07.009.

Montgomery, D.C. (1993). *Introduction to Statistical Quality Control (3ed.)*. Wiley, New York.

Montgomery, D.C., Peck, E.A., and Vining, G.G. (2001). *Introduction to Linear Regression Analysis (3ed.)*. Wiley, New York.

Nomikos, P. and MacGregor, J.F. (1994). Monitoring batch processes using multiway principl analysis. *AIChE Journal*, 40 (8), pp 1361-1375.

Sharma, S. (1996). *Applied Multivariate Techniques*. Wiley, New York.

Wong, J. (2006). Batch PLS analysis and FDC process control of within lot SiON gate oxide thickness variation in sub-nanometer range. *Proc. AEC/APC Symp. XVIII*, Westminster, CO, Sep.

Yue, H.H., Qin, S.J., Markle, R.J., Nauert, C., and Gatto, M. (2000). Fault detection of plasma etchers using optical emission spectra. *IEEE Trans. Semiconduct. Mnauf.*, 13 (3), pp 374-385.

Baldi, P. and S. Brunak. (2001). *Bioinformatics: The Machine Learning Approach*. MIT Press.

Higgins, J.J. (2004). *Introduction to Modern Nonparametric Statistics*, Brooks.

Fault Detection and Diagnosis using Multivariate Statistical Techniques in a Wastewater Treatment Plant. ⋆

D. Garcia-Alvarez * **M.J. Fuente** * **P. Vega** ** **G. Sainz** *

* *Department of Systems Engineering and Automatic Control, University of Valladolid, 47011 Valladolid, Spain*
[dieggar@cta,maria@autom,gresai@eis].uva.es
** *Department of Computer Science and Control, University of Salamanca, ETSII, 37700 Bejar (Salamanca), Spain* pvega@usal.es

Abstract: In this paper Principal Components Analysis (PCA) is used for detecting faults in a simulated wastewater treatment plant (WWTP). Diagnosis tasks are treated using Fisher discriminant analysis (FDA). Both techniques are multivariate statistical techniques used in multivariate statistical process control (MSPC) and fault detection and isolation (FDI) perspectives. PCA reduces the dimensionality of the original historical data by projecting it onto a lower dimensionality space. It obtains the principal causes of variability in a process. If some of these causes change, it can be due to a fault in the process. FDA provides an optimal lower dimensional representation in terms of a discriminant between classes of data, where, in this context of fault diagnosis, each class corresponds to data collected during a specific and known fault. A discriminant function is applied to diagnose faults using data collected from the plant.

Keywords: Fault detection, Fault diagnosis, Statistical process control, Wastewater treatment plant, Discriminant analysis

1. INTRODUCTION

Multivariate statistical methods for the analysis of process data have recently been used successfully for monitoring and fault detection. The safe operation and the production of high quality products are two of the main objectives in industry. Modern control techniques have resolved many problems, but when a special cause occurs in a process, it cannot operate under control. The development of an industrially reliable online scheme for such processes would be a step toward effectiveness and robustness.

Conventional univariate Statistical Process Control (SPC) uses typical control charts, such as Shewhart charts, for monitoring a single variable. When univariate control charts are applied to multivariate systems, with hundreds of variables, the results are improper because, when there is a fault or an abnormality in the operation, several of these charts set off an alarm in a short period of time or simultaneously. This situation is because the process variables are correlated, and a special cause can affect more than one variable at the same time. Multivariate Statistical Process Control (MSPC) uses latent variables instead of every measured variable. All these methods use historical databases to calculate empirical models that describe the system's trend. They are able to extract useful information from the historical data, calculating the relationship between the variables. When a problem appears, it changes the covariance structure of the model and can be detected.

Multivariate statistical approaches, and principal component analysis (PCA) in particular, have been investigated to deal with this problem. Jackson and Mudholkar investigated PCA as a tool of MSPC (Jackson and Mudholkar, 1979) two decades ago. The objective of this approach is to reduce the dimensionality of the original historical data by projecting it onto a lower dimensionality space. PCA finds linear combinations of variables that describe major trends in a data set. Mathematically, PCA is based on an orthogonal decomposition of the covariance matrix of the process variables along the directions that explain the maximum variation of the data. PCA can be studied from two perspectives, one is the cited MSPC, and other is the fault detection and isolation (FDI) perspective, which is discussed by Venkatasubramanian (Venkatasubramanian et al., 2003a,b,c). The author divides the fault detection and diagnosis techniques into three parts: quantitative model-based methods, qualitative models and search strategies and process history-based methods. PCA falls into the third category because it uses historical databases to derive the statistical model (PCA model) (Hwang and Han, 1999; Kourti, 2002; Tien et al., 2004).

The charts most commonly used with PCA techniques are Hotelling statistics, T^2, and the sum of squared residuals, SPE, or Q statistic. The T^2 statistic is a measure of the variation in the PCA model and the Q statistic is a

⋆ This work was supported in part by the national research agency of Spain (CICYT) through the project DPI2006-15716-C02-02 and the regional government of Castilla y Leon through the project VA052A07

measure of the amount of variation not captured by the PCA model.

Once the fault is detected using monitoring techniques, it can be diagnosed by determining the fault region in which the observations are located. The approach used in this paper for fault diagnosis is pattern classification. When the data collected during the *out-of-control* operations have been previously diagnosed, the data can be categorized into separate classes when each class pertains to a particular fault (Chiang et al., 2000).

Fisher discriminant analysis (FDA) is a linear pattern classification method used to find the linear combination of features which best separate two or more classes. It is an empirical method based on observed attributes over the collected examples. FDA provides an optimal lower dimensionality representation in terms of a discriminant between classes of data, where, for fault diagnosis, each class corresponds to data collected during a specific, known fault. FDA has been studied in detail in the pattern classification literature (Duda et al., 2001), but its use for analyzing chemical process data had not been explored until recently (Chiang et al., 2000; He et al., 2005; Fuente et al., 2008).

The purpose of this article is to implement a method for fault detection and diagnosis using multivariate statistical methods and to apply it to a wastewater treatment plant (WWTP). Theoretical aspects of PCA and FDA will be presented and finally the wasterwater treatment plant, the considered faults and the results obtained will be explained and discussed.

2. PRINCIPAL COMPONENT ANALYSIS

Principal component analysis (PCA) is a vector space transformation often used to transform multivariable space into a subspace which preserves maximum variance of the original space in a minimum number of dimensions. The measured process variables are usually correlated to each other. PCA can be defined as a linear transformation of the original correlated data into a new set of uncorrelated data, so, PCA is a good technique to transform the set of original process variables into a new set of uncorrelated variables that explain the trend of the process.

Consider a data matrix $X \in \Re^{n \times m}$ containing n samples of m process variables collected under normal operation. This matrix must be normalized to zero mean and unit variance with the scale parameter vectors $\bar{x}$ and s as the mean and variance vectors respectively. Then next step to calculate the PCA is to construct the covariance matrix R:

$$R = \frac{1}{n-1} X^T X \tag{1}$$

and to perform the SVD decomposition on R:

$$R = V \Lambda V^T \tag{2}$$

where Λ is a diagonal matrix that contains the eigenvalues of R in its diagonal sorted in decreasing order ($\lambda_1 \geq \lambda_2 \geq \ldots \geq \lambda_m \geq 0$). Columns of matrix V are the eigenvectors of R. The transformation matrix $P \in \Re^{m \times a}$ is generated by choosing a eigenvectors or columns of V corresponding to a principal eigenvalues. Matrix P transforms the space of the measured variables into the reduced dimension space.

$$T = XP \tag{3}$$

The columns of matrix P are called *loadings* and the elements of T are called *scores*. Scores are the values of the original measured variables that have been transformed into the reduced dimension space.

Operating in equation (3), the scores can be transformed into the original space.

$$\hat{X} = TP^T \tag{4}$$

The residual matrix E is calculated as:

$$E = X - \widehat{X} \tag{5}$$

Finally the original data space can be calculated as:

$$X = TP^T + E \tag{6}$$

It is very important to choose the number of principal components a, because TP^T represents the principal sources of variability in the process and E represents the variability corresponding to process noise. There are several proposed procedures for determining the number of components to be retained in a PCA model, such as Zumoffen and Basualdo (2007) and Jackson (1991):

a) The SCREE procedure (Jackson, 1991): It is a graphical method in which one constructs a plot of the eigenvalues in descending order and looks for the *knee* in the curve. The number of selected components are the components between the high component and the *knee*. An example of this graph is shown in Fig. 2.
b) Cumulative Percent Variance (CPV) approach Zumoffen and Basualdo (2007). A measure of the percent variance ($CPV(a) \geq 90\%$) captured by the first a principal components is adopted:

$$CPV(a) = \frac{\sum_{i=1}^{a} \lambda_i}{trace(R)} 100 \tag{7}$$

c) Cross validation.

Having established a PCA model based on historical data collected when only common cause variations are present, multivariate control charts based on Hotelling's T^2 and square prediction error (SPE) or Q can be plotted. The monitoring can be reduced to these two variables (T^2 and Q) characterizing two orthogonal subsets of the original space. T^2 represents the major variation in the data and Q represents the random noise in the data. T^2 can be calculated as the sum of the squares of a new process data vector x:

$$T^2 = x^T P \Lambda_a^{-1} P^T x \tag{8}$$

where Λ_a is a squared matrix formed by the first a rows and columns of Λ.

The process is considered *normal* for a given significance level α if:

$$T^2 \leq T_\alpha^2 = \frac{(n^2-1)a}{n(n-a)} F_\alpha(a, n-a) \tag{9}$$

where $F_\alpha(a, n-a)$ is the critical value of the Fisher-Snedecor distribution with n and $n-a$ degrees of freedom

and α the level of significance. α takes values between 90% and 95%.

T^2 is based on the first a principal components so that it provides a test for deviations in the latent variables that are of the greatest importance to the variance of the process. This statistic will only detect an event if the variation in the latent variables is greater than the variation explained by common causes.

New events can be detected by calculating the squared prediction error SPE or Q of the residuals of a new observation. The Q statistic (Jackson and Mudholkar (1979), Jackson (1991)) is calculated as the sum of the squares of the residuals. The scalar value Q is a measurement of *goodness of fit* of the sample to the model and is directly associated with the noise:

$$Q = r^T r \tag{10}$$

with:

$$r = (I - PP^T)x$$

The upper limit of this statistic can be computed as follows:

$$Q_\alpha = \theta_1 \left[\frac{h_0 c_\alpha \sqrt{2\theta_2}}{\theta_1} + 1 + \frac{\theta_2 h_0 (h_0 - 1)}{\theta_1^2} \right]^{\frac{1}{h_0}} \tag{11}$$

with:

$$\theta_i = \sum_{j=a+1}^{m} \lambda_j^i \qquad h_0 = 1 - \frac{2\theta_1 \theta_3}{3\theta_2^2}$$

where c_α is the value of the normal distribution, with α being the level of significance.

When an unusual event occurs and it produces a change in the covariance structure of the model, it will be detected by a high value of Q.

3. FISHER DISCRIMINANT ANALYSIS

For fault diagnosis, data collected from the plant during specific faults are categorized into classes, where each class contains data representing a particular fault. Define n as the number of observations, m as the number of measurement variables, p as the number of classes and n_j as the number of observations in the j^{th} class. The training data for all classes have been stacked into the matrix $X \in \Re^{n \times m}$. The total-scatter matrix is:

$$S_t = \sum_{i=1}^{n} (x_i - \bar{x})(x_i - \bar{x})^T \tag{12}$$

where $\bar{x}$ is the total mean vector whose elements correspond to the means of the columns of X. Let the matrix X_j be defined as the set of vectors x_j which belong to class j, then the within-scatter matrix for class j is given by:

$$S_j = \sum_{x_i \in X_j} (x_i - \bar{x}_j)(x_i - \bar{x}_j)^T \tag{13}$$

where $\bar{x}_j$ is the mean vector for class j:

$$\bar{x}_j = \frac{1}{n_j} \sum_{x_i \in X_j} x_i \tag{14}$$

The within-class-scatter matrix is:

$$S_w = \sum_{j=1}^{p} S_j \tag{15}$$

and the between-class-scatter matrix is:

$$S_b = \sum_{j=1}^{p} n_j (\bar{x}_j - \bar{x})(\bar{x}_j - \bar{x})^T \tag{16}$$

The total-scatter matrix is equal to the sum of the between-scatter matrix and the within-scatter matrix: $S_t = S_b + S_w$. The objective of the first FDA vector, w_1, is to maximize the scatter between classes while minimizing the scatter within classes:

$$\max_{w_1 \neq 0} \frac{w_1^T S_b w_1}{w_1^T S_w w_1} \tag{17}$$

with $w_1 \in \Re^m$. The second FDA vector, w_2, is computed so as to maximize the scatter between classes while minimizing the scatter within classes on all axes perpendicular to the first FDA vector, and so on for the remaining FDA vectors. These vectors are equal to the eigenvectors w_k of the generalized eigenvalue problem:

$$S_b w_k = \lambda_k S_w w_k \tag{18}$$

where the eigenvalues λ_k indicate the degree of separability between the classes. As it is the direction and not the magnitude of w_k which is important, the norm is usually chosen to be $\|w_k\| = 1$. The first FDA vector is the eigenvector associated with the largest eigenvalue and so on.

Then, the linear transformation of the data x from the m-dimensional space to the reduced space a-dimensional generated by the FDA vectors is:

$$z_i = W_a^T x_i \tag{19}$$

where $W_a \in \Re^a$ has the a FDA vectors as columns, and $z_i \in \Re^a$. FDA computes the matrix W_a that as the data $x_1, \ldots, x_n$ for the p classes are optimally separated when projected into the a-dimensional space.

There are several methods to choose the number of FDA vectors. These methods are very similar to PCA selection methods, cited in section 2. For example, cross validation or the SCREE procedure.

In order to diagnose the faults, FDA takes into account data collected during different faulty conditions, and uses a discriminant function that takes into account the similarity between the actual data and the data belonging to each class. An observation is assigned to the class i when the maximum discriminant function value, g_i, satisfies:

$$g_i(x) > g_j(x) \ \forall j \neq i \tag{20}$$

where $g_i(x)$ is the discriminant function for class i given a measured vector $x \in \Re^m$. The discriminant function that minimizes the error rate, when the event v_i occurs (for example, the fault i), is (Duda et al., 2001):

$$g_i(x) = P(v_i|x) \tag{21}$$

where $P(v_i|x)$ is the *a posteriori* probability of x belonging to class i. It can be shown that identical classification occurs when the equation (21) is replaced by:

$$g_i(x) = \ln p(x|v_i) + \ln P(v_i) \tag{22}$$

Using the Bayes' rule, considering that the data for each class are normally distributed and characterizing the data to this case, i.e., considering $W_a \in \Re^{m \times a}$ containing the eigenvectors $w_1, w_2, \ldots, w_a$ computed from equation (18), the discriminant function for each class can be derived as:

$$g_j(x) = -\frac{1}{2}(x - \bar{x}_j)^T W_a (\frac{1}{n_j - 1} W_a^T S_j W_a)^{-1} W_a^T (x - \bar{x}_j) + \ln(p_j) - \frac{1}{2} \ln[\det(\frac{1}{n_j - 1} W_a^T S_j W_a)] \quad (23)$$

where S_j, $\bar{x}_j$ and n_j are defined in equations (13) and (14) respectively.

4. APPLICATION

The approach presented in this paper has been tested in a simulated wastewater treatment plant (WWTP). This plant is based on the COST benchmark (Copp, -; Alex et al., 2008). This benchmark was developed for the evaluation and comparison of different activated sludge wastewater treatment control strategies. The model is implemented using MATALAB© and SIMULINK©.

Fig. 1 shows an overview of this plant. It is composed of a two-compartment activated sludge reactor consisting of two anoxic tanks followed by three aerated tanks. This type of plant combines nitrification with predenitrification in a configuration that is usually built for achieving biological nitrogen removal in full-scale plants. The reactor is followed by a secondary settler. The settler is modeled as a 10 layer non-reactive unit. The 6^{th} layer is the feed layer. Table 1 shows the physical parameters of the plant.

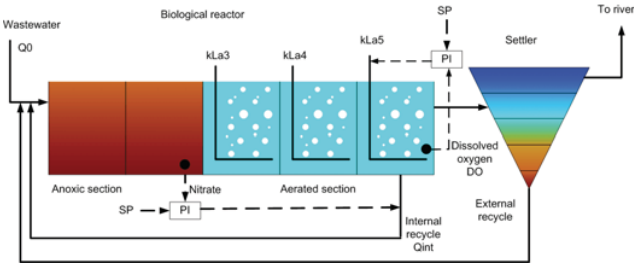


Fig. 1. General overview of the wastewater treatment plant (WWTP)

Table 1. Physical parameters

Elements	Values	Units
Volume - Anoxic section	2000 (2 × 1000)	m^3
Volume - Aerated tank	4000 (3 × 1333)	m^3
Volume - Settler (10 layers)	6000	m^3
Area - Settler	1500	m^2
Height - Settler	4	m

The influent used was the dry influent data file Copp (-). In this file, the variation of influent flow is between $15000 - 35000$ m^3/d. The plant, as Fig. 1 shows, has two reflux: external refluxes, from settler to input, which is approximately equal to the influent flow, and internal reflux, from the last aerated tank to input, which is approximately equal to three times the influent flow, but which is a controlable variable.

The objective of the control strategy is to control the dissolved oxygen level in the aerated reactor by manipulation of the oxygen transfer coefficient ($K_L a_5$) and to control the nitrate level in the anoxic tank by manipulation of the internal recycle flow rate. The controllers are of PI type. Tab. 2 shows the principal controller settings.

Table 2. Controllers settings

Variables	Oxygen loop	Nitrate loop
Controller type	PI	PI
Controlled variable	DO $[g/m^3]$	S_{NO} $[gN/m^3]$
Manipulated variable	$K_L a5$ $[1/hr]$	Q_{int} $[m^3/d]$
Setpoint	2 $[g/m^3]$	1 gN/m^3

The model of the plant is formed by 13 state variables. The variables involved are concentrations of:

1. Alkalinity (S_{ALK}).
2. Soluble biodegradable organic nitrogen (S_{ND}).
3. Ammonia nitrogen (S_{NH}).
4. Nitrate (S_{NO}).
5. Dissolved oxygen (S_O).
6. Readily biodegradable substrate (S_S).
7. Active autotrophic biomass ($X_{B,A}$).
8. Active heterotrophic biomass ($X_{B,H}$).
9. Particulate biodegradable organic nitrogen (X_{ND}).
10. Particulate products from biomass decay (X_P).
11. Slowly biodegradable substrate (X_S).
12. Particulate inert organic matter (X_I).
13. Soluble inert organic matter (S_I).

In this case, three faults have been considered. They are not sensors or actuators faults, they are faults in the process. The faults considered are:

- **Toxicity shock.** This fault is due to a reduction in the normal growth of heterotrophic organisms. This type of fault can be produced by toxic substances in the water coming from textile industries or pesticides. This fault is simulated by reducing the maximum heterotrophic growth rate (μ_H).
- **Inhabitation.** This fault can be produced by hospital waste that can contain bactericides, or metallurgical waste that can contain cyanide. This type of fault is due to a reduction in the normal growth of the heterotrophic organisms and an increase in the decay factor of this type of organisms. This fault is similar to toxicity shock but is more drastic. In this case, the fault is caused by reducing the maximum heterotrophic growth rate (μ_H) and increasing the heterotrophic decay rate (b_H).
- **Bulking.** This type of fault is produced by the growth of filamentous microorganisms in the active sludge. This phenomenon causes the impossibility of decantation in the settler. To simulate this fault the settling velocity in layer (v_{sj}) is reduced.

More information about these parameters and mathematical models can be consulted in Copp (-). But in this example the benchmark has been modified in order to introduce the fault parameters.

There are several groups working on fault detection in wastewater treatment plants using PCA (Rosen and Lennox, 2001) or using other fault detection approaches (Genovesi et al., 2000).

Using this dynamic model, the results were obtained in steady state. For this, the plant model has to simulate $100-150$ days in open-loop configuration and determines this steady state. Then, the simulation in closed-loop is simulated for 14 days and faults are caused on the 7^{th} day. The samples for monitoring experiments were taken 100 times per day.

The selected variables to calculate principal components analysis (PCA) and Fisher discriminant analysis (FDA) are the first eleven state variables and the effluent flow rate (Q_0). The concentration of particulate inert organic matter (X_I) and soluble inert organic matter (S_I) are not relevant to this study (Tomita et al., 2002).

The number of principal components, calculated using the CPV approach with 95% maximum variance level, are five, but Fig. 2 shows that seven principal components can be a better option because they capture more variability of the process.

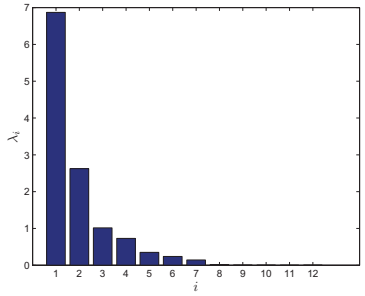


Fig. 2. SCREE graph for principal component selection

The process monitoring under toxicity shock fault can be seen in Fig. 3. The thresholds of both statistics T^2 and Q rise when the fault occurs. In this case, the Q statistic detects this fault better than the T^2 statistic, as this figure shows.

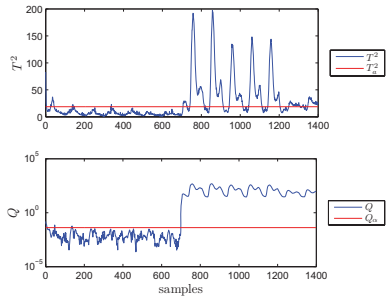


Fig. 3. Toxicity shock fault detection. Logarithmic scale for Q statistic.

The inhabitation fault detection is more effective than the detection of the toxicity shock fault because this type of fault is more drastic, as can be seen in Fig. 4. Finally, the bulking fault detection using PCA is shown in Fig. 5.

The number of selected FDA vectors for fault diagnosis tasks was two using the the SCREE graph method. Fig. 6 shows the discriminant functions (g_i, eq. 23) when a toxicity shock fault has been caused. The solid line corresponds

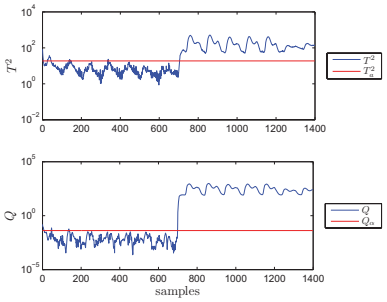


Fig. 4. Inhabitation detection. Logarithmic scale for T^2 and Q statistics.

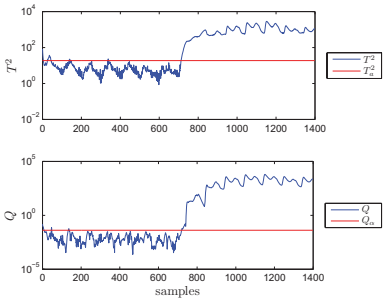


Fig. 5. Bulking fault detection. Logarithmic scale for T^2 and Q statistics.

to the discriminant function for toxicity shock fault, the dotted line corresponds to the discriminant function for inhabitation fault and the dashed line corresponds to the discriminant function for the bulking fault. In this case, once the fault has been detected (7^{th} day) the discriminant function for the toxicity shock fault is greater than the rest of the discriminant functions, so the fault is correctly diagnosed. The experimented faults used to find results are different from the considered faults used in the training data.

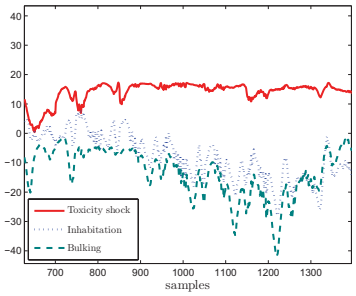


Fig. 6. Toxicity shock fault diagnosis

Fig. 7 shows the discriminant function graphs in the case where the inhabitation fault has occurred. In this situation, the discriminant function for the inhabitation fault is the greatest, so the fault is correctly diagnosed.

Finally, Fig. 8 shows the evolution of the discriminant function for the bulking fault. In this case, the evolution for the bulking fault is always greater than for the rest of the discriminant faults.

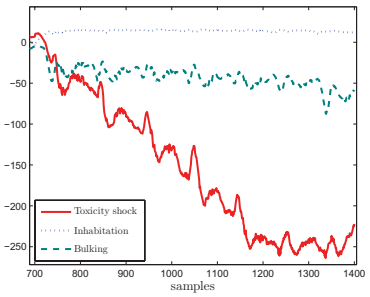


Fig. 7. Inhabitation fault diagnosis

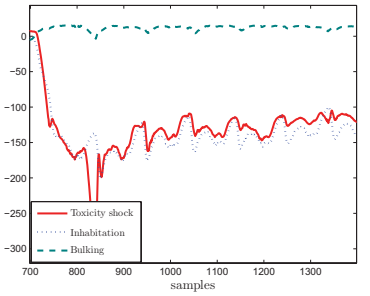


Fig. 8. Bulking fault diagnosis

5. CONCLUSIONS

This paper proposes an approach to deal with the fault detection and diagnosis using statistical techniques, concretely, the principal component analysis (PCA) is used in detection tasks and the Fisher discriminant analysis (FDA) is implemented in diagnosis tasks.

The approach has been proved in a simulated wastewater treatment plant (WWTP) based on the COST benchmark. The considered faults are critical process faults that affect some plant parameters. Data are collected from the plant for normal conditions in order to calculate the PCA model and the thresholds of the T^2 and Q statistics, used to detect the faults. Data for different classes (parameter faults) are also collected to calculate the FDA models for diagnosis. The used approach shows good results because the faults was detected and correctly diagnosed.

A useful update to this work can be to obtain data when two or more faults occur simultaneously. New discriminant functions can be calculated using this data and these new situations could be diagnosed.

ACKNOWLEDGEMENTS

The authors would like to express their gratitude to the department of industrial electrical engineering and automation of Lund University for the implementation of BSM1 model.

REFERENCES

Alex, J., Benedetti, L., Copp, J., Gernaey, K., Jeppsson, U., Nopens, I., Pons, M., Rieger, L., Rosen, C., Steyer, J., Vanrolleghem, P., and Winkler, S. (2008). *Benchmark Simulation Model no. 1 (BSM1)*. Dept. of Industrial Electrical Engineering and Automation. Lund University.

Chiang, L., Russell, E., and Braatz, R. (2000). *Fault Detection and Diagnosis in Industrial Systems*. Springer, Nueva York.

Copp, J. (-). *The COST Simulation Benchmark: Description and Simulator Manual (a product of COST Action 624 & COST Action 682)*. European Cooperation in the field of Scientific and Technical Research.

Duda, R., Hart, P., and Stork, D. (2001). *Pattern Clasiffication*. Wiley, New York, 2nd edition.

Fuente, M., Garcia, G., and Sainz, G. (2008). Fault diagnosis in a plant using fisher discriminant analysis. *16th Mediterranean Conference on Control and Automation Congress Centre, Ajaccio, France*, 53–58.

Genovesi, A., Harmand, J., and Steyer, J. (2000). Integrated fault detection and isolation: Application to a winery's wastewater treatment plant. *Applied Intelligence*, 13, 59–76.

He, Q., Qin, S., and Wang, J. (2005). A new fault diagnosis method using fault directions in fisher discriminant analysis. *AIChE Journal*, 51(2), 555–571.

Hwang, D. and Han, C. (1999). Real-time monitoring for a process with multiple operating modes. *Control Engineering Practice*, 7, 891–902.

Jackson, J. (1991). *A user's guide to principal components*. Wiley.

Jackson, J. and Mudholkar, G. (1979). Control procedures for residuals associated with principal component analysis. *Technometrics*, 21, 341–349.

Kourti, T. (2002). Multivariable dynamic data modeling for analysis and statistical process control of batch processes, start-ups and grade transitions. *Journal of Chemometrics*, 17, 93–109.

Rosen, C. and Lennox, J. (2001). Multivariate and multiscale monitoring of wastewater treatment operation. *Water research*, 35, 3402–3410.

Tien, D., Lim, K., and Jun, L. (2004). Compartive study of pca approaches in process monitoring and fault detection. *The 30th annual conference of the IEEE industrial electronics society*, 2594–2599.

Tomita, R., Park, S., and Sotomayor, O. (2002). Analysis of activated sludge process using multivariate statistical tools - a pca approach. *Chemical Engineering Journal*, 90, 283–290.

Venkatasubramanian, V., Rengaswamy, R., Kavuri, S., and Yin, K. (2003a). A review of process fault detection and diagnosis. part i: Quantitative model-based methods. *Computers & Chemical Eng.*, 27, 291–311.

Venkatasubramanian, V., Rengaswamy, R., Kavuri, S., and Yin, K. (2003b). A review of process fault detection and diagnosis. part ii: Qualitative models and search strategies. *Computers & Chemical Eng.*, 27, 313–326.

Venkatasubramanian, V., Rengaswamy, R., Kavuri, S., and Yin, K. (2003c). A review of process fault detection and diagnosis. part iii: Process history based methods. *Computers & Chemical Eng.*, 27, 327–346.

Zumoffen, D. and Basualdo, M. (2007). From large chemical plant data to fault diagnosis integrated to decentralized fault tolerant control. *Industrial & Eng. Chemistry Research*, 47, 1201–1220.

On the structure determination of a dynamic PCA model using sensitivity of fault detection

Mohamed Guerfel * **Kamel Ben Othman** *
Mohamed Benrejeb *

* *LARA Automatique, ENIT, BP 37, 1002 Tunis Belvédère, Tunisia (e-mail: guerfel_mohamed@yahoo.fr, kamel.benothman@enim.rnu.tn, mohamed.benrejeb@enit.rnu.tn)*

Abstract: This work proposes a dynamic PCA modeling method for dynamical non-linear processes. This method uses fault free data to construct data matrix used to compute the correlation matrix and faulty system data in order to fix the dynamic PCA model parameters (the time-lag and the number of principal components). It is shown that the sensitivity of dynamic PCA-based fault detection depends on the parameters used in the model. This method is tested on a three serial interconnected tanks and subject to fluid circulation faults in its pipes.

Keywords: Process and control monitoring; Modeling; Static PCA; Dynamic PCA; Time-lag; Number of components; System fault detection.

1. INTRODUCTION

The use of multi statistical process control tools also known as MSPC became frequent for the modeling, control and diagnosis of complex and over-instrumented processes (chemicals, microelectronics, pharmaceutical..., see Venk (2003)). Static principal component analysis (PCA) is one among the most popular statistical methods, it was used successfully as a modeling tool for statical and slow dynamics processes in linear or non-linear cases, (see Qin (2003)). The extension of PCA for the dynamical modeling, called DPCA, was proposed in Ku (1995). Other work tackled this subject, like in Lee (2004), Li (2003), Mina (2007), Treasure (2004), Xie (2006). In all methods presented in scientific litterature, the model used as reference in the diagnosis procedure is obtained via the minimization of a criterion depending on the nominal data of the process. However, the obtained model can be inadequate for changes detection purposes since the minimized criterion does not necessarily maximize the changement impact of the process on the computed model (see Tamura (2007), Kano (2002)). Many changement types can affect the process, among them one distinguishes : sensors/actuators failures (see Huang (2000)), performance degradation (see Kano (2002)), operating point changes (see Zhao (2004)) and process structure modification or "system fault" (see Huang (2007)). These changes can be highlighted by various statistical tests chosen according to the changement type to be detected. For further details on these tests (also called residuals), the reader can consult Harkat (2006), Kano (2001), Singhal (2005), Guerfel (2008). This work proposes a modeling method of dynamic, linear or non-linear processes via DPCA. This method jointly uses nominal process data to build the correlation matrix to diagonalize and system fault type data to fix the time-lag and the principal component number to retain for the DPCA model. The paper is divided into the following sections. Section 2 recalls shortly the static PCA modeling and its structural parameter. Section 3 defines the dynamic PCA modeling and its structural parameters. The different changes which can affect a process and the statistical test used in this work to detect the system fault type are defined in section 4. The proposed modeling method permitting the choice of the time-lag and the number of principal component to retain for the DPCA model in the case of dynamical non-linear process is presented in section 5. Section 6 illustrates the application of the method on a three serial tanks subject to fluid circulation faults in their pipes. Finally, the last section provides a concluding summary of this work.

2. STATIC PRINCIPAL COMPONENT ANALYSIS

For the vector $z(k) = [z_1(k)\, z_2(k)\, \ldots\, z_m(k)]^t$, scaled to zero mean and unity variance and containing the m observed inputs/outputs of the process in the instant k, the data matrix Z_N resulting from the juxtaposition of $z(k)$ in different instants is written :

$$Z_N = [\, z(k) \cdots z(k+N-1)\,]^t \tag{1}$$

The subscripts N designates the number of observations used in the construction of the matrix Z_N.
Modeling a process via static PCA consists in seeking an optimal linear transformation (with respect to a variance criterion) of the original data matrix Z_N into a new one called T and defined as follows :

$$T = Z_N P \quad ; \quad T = [\, t_1 \cdots t_m\,] \in \mathbb{R}^{N\times m} \tag{2}$$

The vectors $t_q \in \mathbb{R}^N$, $q \in \{1, \ldots, m\}$, called principal components are uncorrelated and arranged in the decreasing variance order. The column vectors p_q of the matrix P

represent the eigenvectors corresponding to the eigenvalues λ_q obtained from the diagonalization of the correlation matrix Σ of Z_N :

$$\Sigma = P\Lambda P^t \quad ; \quad PP^t = P^tP = I_m \tag{3}$$

the notation $\Lambda = diag(\lambda_1 \dots \lambda_m)$ designates the diagonal matrix of eigenvalues arranged in the decreasing magnitude order $\lambda_1 \geq \lambda_2 \geq \dots \geq \lambda_m$. With the triple partitioning :

$$\Lambda = \begin{bmatrix} \hat{\Lambda} & 0 \\ 0 & \tilde{\Lambda} \end{bmatrix}, \quad P = \left[\, \hat{P} \middle| \tilde{P} \,\right], \quad T = \left[\, \hat{T} \middle| \tilde{T} \,\right] \tag{4}$$

The data matrix can be decomposed in the following form :

$$Z_N = \hat{Z}_N + E_N \text{ with } \hat{Z}_N = Z_N\hat{C} \ ; \ E_N = Z_N\tilde{C} \tag{5}$$

The matrices $\hat{C} = \hat{P}\,\hat{P}^t$ and $\tilde{C} = I_m - \hat{C}$ form the static PCA model of the process (for further details see Jollife (2003)). The matrices $\hat{Z}_N$ and E_N represent, respectively, the modeled and the non modeled variations of Z_N from ℓ components ($\ell < m$). The first ℓ eigenvectors forming the matrix $\hat{P} \in \mathbb{R}^{m\times\ell}$ constitute the representation space whereas the last $(m-\ell)$ eigenvectors forming the matrix $\hat{P} \in \mathbb{R}^{m\times\ell}$ constitute the residual space.
The identification of the static PCA model thus consists in estimating its parameters by an eigenvalue/eigenvector decomposition of the matrix Σ and determining its structural parameter which is the number of principal components ℓ to retain. An incorrect choice (too large or too small) of ℓ could mask the changes occurring in the modeled process or gives false alarms which affect the change detection procedure (see Qin (2003)). Many methods were proposed to fix ℓ in the static PCA model. The reader can find more details in Valle (1999). Most of these methods are heuristic and give a subjective number ℓ (see Harkat (2005)). In order to mitigate the disvantages of the heuristic methods, Qin and Dunia have proposed to fix ℓ via the minimization of a criterion called VNR which represents the variance of the reconstruction error of the process variables (see Qin (2000)). However, it is noted that the VNR criterion underestimates the number ℓ exact to retain in real applications cases (see Valle (1999)). All the methods aiming at the determination of ℓ seek to find its theoretical or exact value called (ℓ_{th}) which represents the theoretical number of linear or quasi-linear relations existing between the different components of $z(k)$. These methods are sensitive to the signal noise ratio and depend on the nature of the process non linearity. It is also noted that ℓ can be different from ℓ_{th} in the case of models built for diagnosis purposes provided that the static PCA model (constructed with ℓ components) can detect changes (see Frank (2000)). From this idea was born a new process modeling method via static PCA. Proposed in Tamura (2007), this method uses nominal process data to build the correlation matrix which will be diagonalized and faulty data in order to fix ℓ.

3. DYNAMIC PRINCIPAL COMPONENT ANALYSIS

Dynamic principal component analysis proposed in Ku (1995) and known as DPCA aims at finding dynamical linear relations between the process variables. The principle of this method is identical to the static PCA. Starting from a scaled to zero mean and unity variance data vector $z^d(k) = [z^t(k)\ z^t(k-1)\ \dots\ z^t(k-s)]$, where s designates the used time-lag, the data matrix $Z_N^d\,(k,s) \in \mathbb{R}^{N\times m(s+1)}$ is built as follows :

$$Z_N^d\,(k,s) = \begin{bmatrix} z^t\,(k) & \dots & z^t\,(k-s) \\ z^t\,(k+1) & \dots & z^t\,(k-s+1) \\ \vdots & \ddots & \vdots \\ z^t\,(k+N-1) & \dots & z^t\,(k+N-1-s) \end{bmatrix} \tag{6}$$

with $N > m(s+1)$ and $k > s$.
The correlation matrix obtained from $Z_N^d\,(k,s)$, noted Σ_d, is computed and diagonalized in order to obtain the eigenvectors and the eigenvalues matrices noted respectively P_d and Λ_d. Each one of these two matrices is divided into two parts the first corresponding to the representation space $(\hat{\Lambda}_d,\ \hat{P}_d)$ and the second corresponding to the residual space $(\tilde{\Lambda}_d,\ \tilde{P}_d)$. The principal components vector noted $t^d \in \mathbb{R}^{1\times m(s+1)}$, can be computed in an instant k as follows :

$$t^d(k) = \left[\, \hat{t}^d(k) \mid \tilde{t}^d(k) \,\right] = z^d(k)P_d = z^d(k)\left[\, \hat{P}_d \mid \tilde{P}_d \,\right] \tag{7}$$

The structural parameters in DPCA modeling are the number of principal components ℓ and the time-lag s. The number ℓ can be fixed via the methods used in static PCA after the choice of s which is a very delicate problem. The modeling of data obtained from dynamic process via static PCA constructs an approximate static model of the real process and does not reveal its exact structure (see Ku (1995)). It is possible to detect modifications in dynamical processes via static PCA as in Harkat (2006) and Sharmin (2008), but the theoretical bases of the method are lost since the principal components are no longer uncorrelated and do not follow a normal multivariate statistical distribution. In this case, it will be very difficult to detect small changes in the process parameters as long as the variation domain of the process variales remain the same before and after the change. For a well chosen time-lag $s = s_{min}$, all the static and dynamic relations ruling the process will be represented by the last eigenvectors corresponding to the smallest eigenvalues of Σ_d computed from $Z_N^d\,(k, s_{min})$. Taking a time-lag s higher than s_{min} in the construction of $Z_N^d\,(k,s)$ used for the computation of the DPCA model will not bring any supplementary information but will add redundant relations which were obtained from the construction of the DPCA model using the matrix $Z_N^d\,(k, s_{min})$ (see Ku (1995)). Many methods were proposed for the choice of s. They seek to find the theoritical $s = s_{min}$, most of them are heuristic as in Ku (1995) or resulting from the identification techniques as AIC, see Akaike (1974), Larimore (1990), Li (2003) and MDL, see Simoglou (2002), Rissanen (1978) which privilege the approximation of the data matrix. None of those methods was built in the purpose of minimization of s compared to fault detection.

4. STATISTIC USED FOR SYSTEM FAULT DETECTION

The physical processes are subject to changes in their operating conditions. In the case of non stationary processes, these changes can be sensors/actuators failures (see Huang (2000)), operating point changes, performance degradation or process structure modification. The operating point

changes are characterized by an augmentation in the mean of one or many inputs (see Zhao (2004)). The process performance degradation can be expressed as an augmentation in the variance of one or many process variables under the hypothesis of independent and identically distributed noise (see Kano (2002)). The process structure modifications known as "system fault" appear as changes in the structure or a modification in its model parameters (see Huang (2007)). All these changes can be highlighted by various statistical tests chosen according to the changement type to be detected. Only the system fault type is included in this work. The best indices for the detection of such modification are the ones based on the residual space (see Guerfel (2008)). For this reason, one proposes the use of the D_i statistic, $i = 1, 2, \ldots, (m(s+1) - \ell)$, which is defined in Harkat (2006) as the sum of squared last principal components. This statistic is computed every instant k :

$$D_i(k) = \sum_{j=m(s+1)-i+1}^{m(s+1)} \left(\tilde{t}_j^d(k)\right)^2 \tag{8}$$

The variable $\tilde{t}_j^d(k)$ designates the j^{th} principal component obtained at the instant k. The index D_i represents an SPE (see box (1954)) computed from a DPCA model with $(m(s+1)-i)$ components, its detection threshold $\tau_{i,\alpha}^2$, can be computed in the following way :

$$\tau_{i,\alpha}^2 = g^{(i)} \chi^2_{h^{(i)},\alpha} \tag{9}$$

The notation χ^2 designates the *Chi*-square distribution, α designates the used confidence limit, $g^{(i)}$ and $h^{(i)}$ are defined as follows :

$$g^{(i)} = \frac{\sum_{j=m(s+1)-i+1}^{m(s+1)} \lambda_j^2}{\sum_{j=m(s+1)-i+1}^{m(s+1)} \lambda_j} \qquad h^{(i)} = \frac{\left(\sum_{j=m(s+1)-i+1}^{m(s+1)} \lambda_j\right)^2}{\sum_{j=m(s+1)-i+1}^{m(s+1)} \lambda_j^2} \tag{10}$$

The quantities $\lambda_j,\ j \in \{m(s+1) - i + 1, \ldots, m(s+1)\}$, designate the j^{th} eigenvalues of Σ_d. A system fault is detected if the D_i index is higher than its threshold $\tau_{i,\alpha}^2$.

5. PROPOSITION FOR DYNAMICAL PROCESSES MODELING

The proposed method is similar to the one defined in Tamura (2007) and called MDM abreviation of *Multi Dimensionnal Monitoring*. Contrary to the MDM, the proposed method allows not only the choice of ℓ but also the choice of the minimum time-lag s to retain for the data matrix $Z_N^d(k, s)$ used in the construction of the DPCA process model. The principle of the method is the following :

(1) Begin method
(2) Initialization $S_{init} = 0$ and $\ell = 0$
(3) Build $Z_N^d(k, S_{init})$, Compute Σ_d, P_d and Λ_d
(4) Compute D_i from system fault data for i varying from 1 to the number of column in $Z_N^d(k, s)$ minus 1
(5) If the fault is detected with any of D_i then go to step 7 else go to step 6
(6) $S_{init} = S_{init} + 1$ go to step 3
(7) $s = S_{init}$ and ℓ is equal to the difference between the number of column in $Z_N^d(k, s)$ and the largest value of i which permits the fault detection.
(8) End method

The disadvantage of the proposed method lies in the fact that a knowledge of information on the system fault is necessary to ensure the choice of structural parameters (s and ℓ) to be retained for the DPCA model. From another point of view, if information on the system fault is available, this method becomes very attractive because it determines the simplest model allowing the system fault detection. Figure (1) summarizes the algorithm of the method in the case of a single system fault "j" affecting the modeled process.

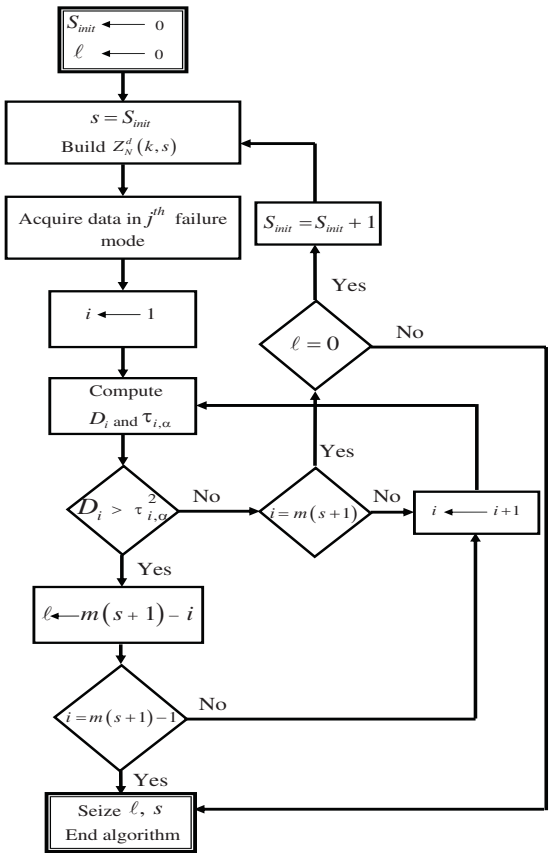


Fig. 1. Algorithm of the proposed method for the determination of s and ℓ in the DPCA model

In order to supress false alarms, the process is considered in failure mode ($D_i > \tau_{i,\alpha}^2$), if D_i has shown eight succeeding values larger than $\tau_{i,\alpha}^2$. The value "eight" is determined in an empirical way and must be adjusted according to the treated application.

6. APPLICATION IN THE MODELING OF THREE TANK SYSTEM

The modeled process illustrated in figure (2), is formed by three identical serial tanks. It contains two inputs : flows q_1, q_2 and three outputs H_1, H_2 and H_3 representing respectively the heights in the first, second and third tanks.

These tanks are interconnected at the bottom by pipes. Two valves V_3 and V_2, separating respectively tank 2 from tank 3 and tank 2 from the outside are introduced in order to model the flows perturbations in the pipes. For a sampling period equal to one second, the discrete process equations are :

$$\begin{cases} H_1(k) = A^{-1}\left(q_1(k) + q_{31}(k) - q_{10}(k)\right) + H_1(k-1) \\ H_2(k) = A^{-1}\left(q_2(k) - q_{23}(k) - q_{20}(k)\right) + H_2(k-1) \\ H_3(k) = A^{-1}\left(q_{23}(k) - q_{31}(k)\right) + H_3(k-1) \\ q_{10}(k) = K_1\sqrt{H_1(k)} \\ q_{20}(k) = K_2\sqrt{H_2(k)} \\ q_{31}(k) = K_{31} f\left(H_3(k) - H_1(k)\right) \\ q_{23}(k) = K_{23} f\left(H_2(k) - H_3(k)\right) \end{cases} \tag{11}$$

where A equal to $0.01539\,m^2$, designates the tank section. The constants K_1, K_2, K_{31} and K_{23} respectively equal to $1.816\,e^{-4}$, $9.804\,e^{-5}$, $1.005\,e^{-4}$ and $7.804\,e^{-5}$ are the process characteristics. The term $f(.)$ designates a non linear function defined as follows :

$$f(x) \triangleq sign(x)\sqrt{|x|} \tag{12}$$

The measured process variables are the inputs z_1^b, z_2^b and

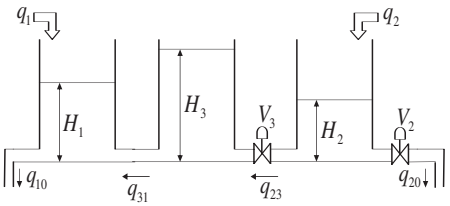


Fig. 2. Three tanks system

the outputs z_3^b, z_4^b and z_5^b. Theses measures are related in the instant k to the physical values via the following equations :

$$\begin{cases} z_1^b(k) = H_1(k) + \varepsilon_1(k) \\ z_2^b(k) = H_2(k) + \varepsilon_2(k) \\ z_3^b(k) = H_3(k) + \varepsilon_3(k) \\ z_4^b(k) = q_1(k) + \varepsilon_4(k) \\ z_5^b(k) = q_2(k) + \varepsilon_5(k) \end{cases} \tag{13}$$

The quantities $\varepsilon_r(k)$, $r \in \{1, \ldots, 5\}$ designate gaussian centered measurement noise. Its standard deviation is equal to 3% of that of the entries. The flows q_1 and q_2 are expressed in m^3/s. They are chosen to be random durations crenels with variable amplitudes respectively in $[3.20, 6.71] \times 10^{-5}$ for q_1 and in $[5.73, 9.57] \times 10^{-5}$ for q_2. The tanks initial heights are expressed in meter. Their values are 0.147, 0.276 and 0.195 respectively for the first, second and third tank. The system is firstly simulated under nominal operation during 4000 samples. After centering and reducing the inputs/outputs measures, the vector z is built at each instant k as follows :

$$z(k) = [\, z_1(k)\ z_2(k)\ z_3(k)\ z_4(k)\ z_5(k)\,]^t \tag{14}$$

where $z_r(k)$ designates the centered and reduced value of $z_r^b(k)$. The data matrix is constructed via (1). It will be used in the computation of the matrices Λ and P.
In the dynamical case, the vector z^d is constructed in an instant k for a time-lag s using time-lagged vectors z obtained in the static case as following :

$$z^d(k) = \left[\, z^t(k)\ z^t(k-1)\ \ldots\ z^t(k-s)\,\right] \tag{15}$$

The data matrix $Z_N^d(k,s)$ is built via (6). It will be used for the computation of the matrices Λ_d and P_d.
Figure 3 shows the process scree plot for a time-lag s respectively equal to zero, one and two. The eigenvalues of Σ (built for $s = 0$) are not null and do not indicate the presence of any linear or quasi linear relation between the measured process variables at the same instant. The last three eigenvalues of Σ_d built for $s = 1$ are quasi null and show the existance of three quasi linear relations verified by the measured process variables between two sampling instants. The correlation matrix Σ_d built for $s = 2$ shows six quasi null eigenvalues. They indicates the presence of six quasi linear relations verified by the measured process variables between three sampling instants. The approximation of the non linear relations verified by the measured process variables may be better if one uses time-lags higher than two but the computation complexity will increase as well.

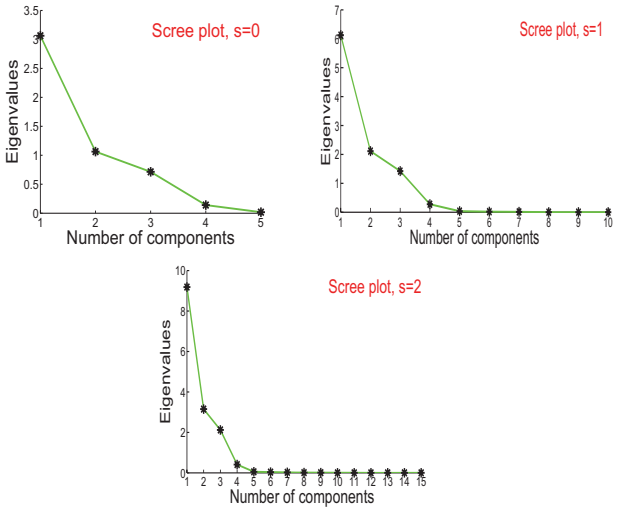


Fig. 3. Scree plots for $s = 0$, $s = 1$ and $s = 2$

Twenty measures representing a system fault are generated in order to fix the time-lag s used for the construction of $Z_N^d(k,s)$. This system fault represents a variation of 1.1 of K_2 from its nominal value. The application of the algorithm in section 5 gives that $s = 1$ and $\ell = 7$ are sufficient to detect such a fault.
A second simulation shown in figure 4 is realized during 4000 instants and perturbations in flows circulation are introduced by varying K_2 in the following way. For the first 1000 instants, K_2 is equal to its nominal value. In the second 1000 instants, K_2 is equal to $1.1\times$ its nominal value. In the third 1000 instants, K_2 is equal to $1.2\times$ its nominal value. For the latest 1000 instants, K_2 is equal to $1.3\times$ its nominal value.
The evolution of the statistics D_i, $i \in \{i, \ldots, 4\}$, respectively for $s = 0$ and $s = 1$ in the case of the second simulation are shown in figure 5 and 6. On one hand, the flows perturbations are not detectable with the statical PCA model. The statistics D_i obtained from its application on

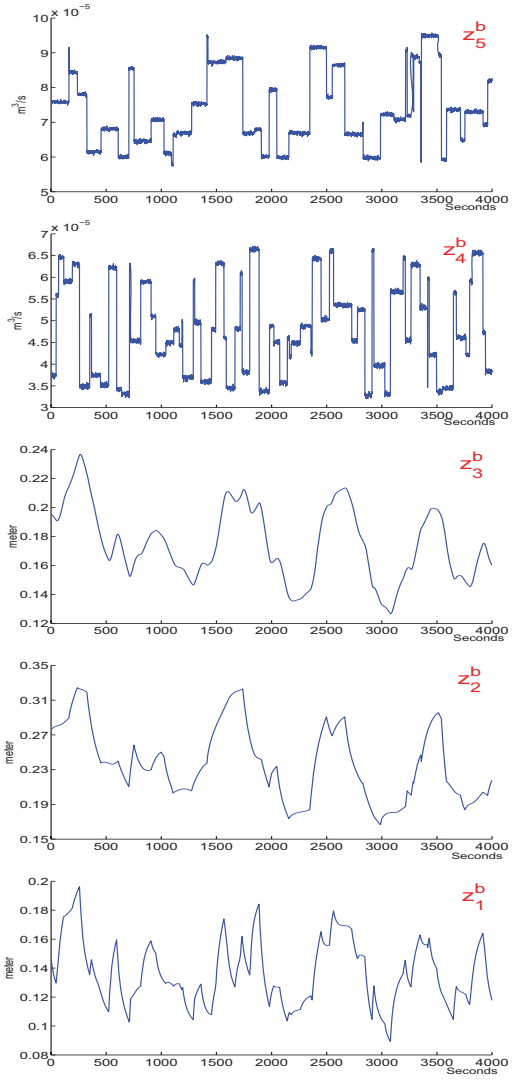


Fig. 4. Evolution of flows z_5^b, z_4^b and heights z_3^b, z_2^b and z_1^b in the second simulation

the current process data are always under their thresholds (figure 5) excepting some aberrant values. On the other hand, the flows perturbations are well detected with a DPCA built with $s = 1$. The statistics D_i, $i \in \{1, \dots 3\}$, obtained from its application on the current process data allow the detection of all the system faults that were simulated. The maximal dimension of the residual space that can detect the perturbations is equal to three. The time-lag s used to construct the data matrix from which the DPCA model is built depending on the magnitude of the system fault to be detected. Increasing the value of s may lead to a better linear approximation of the real non linear relations existing between the time-lagged process measurements which can reduce the magnitude of the system fault to be detected.

7. CONCLUSION

The proposed method permits the estimation of the time-lag s and the choice of the principal component number ℓ used in the process modeling via DPCA. Contrary to the

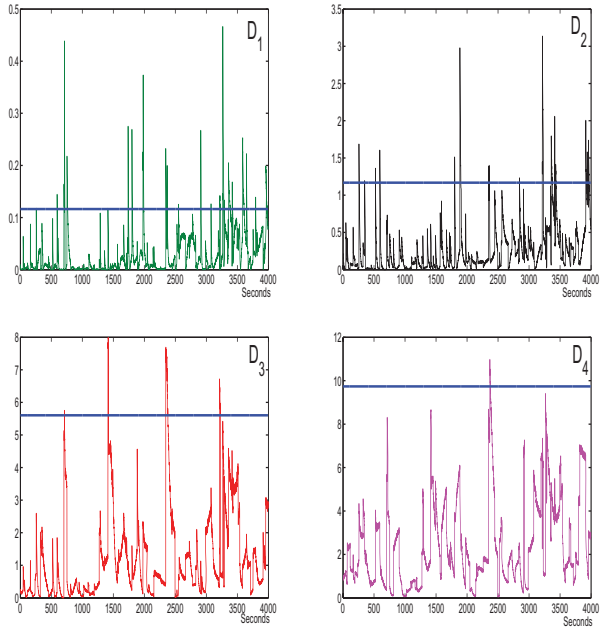


Fig. 5. Evolution of $D_i, i \in \{1, \dots, 4\}$ in the case of $s = 0$

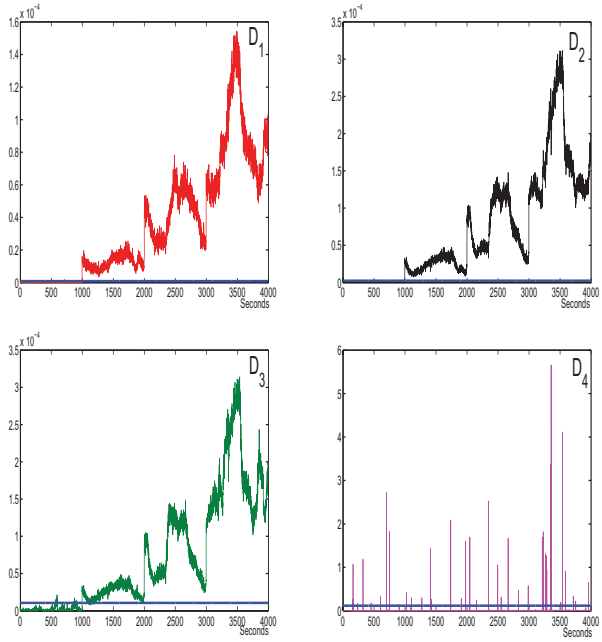


Fig. 6. Evolution of $D_i, i \in \{1, \dots, 4\}$ in the case of $s = 1$

majority of the existing methods which use data gathered during nominal operation conditions to estimate s and ℓ, the proposed method uses data representing nominal process operating condition to build the data matrix which is used in the computation of DPCA model and other process data belonging to the system fault type to compute the structural parameters s and ℓ. The suggested method proves to be interesting if information relating to the system fault mode is a priori available. In this case, the method determines the least complex model allowing the detection of the considered system fault. Built around a

particular operating point, this method can be sensitive to operating point changes. In the abscence of a system fault, the obtained model presents a risk of generating false alarms due to a shift of the current process variables operating point. The extension of the method to multiple system fault cases and the minimization of the false alarms due to the operating point shift will be considered in forthcoming works.

REFERENCES

H. Akaike. A new look at the statistical model identification. *IEEE Transactions on Automatic Control*, volume 19, pages 716–723, 1974.

G. E. P. Box. Some theorems on quadratique forms applied in the study of analysis of variance problems : Effect of inequality of variance in one-way classification. *The Annals of Mathematical Statistics*, volume 25, pages 290–302, 1954.

P.M. Frank, E. A. Garcia, B. Köppen-Seliger. Modelling for fault detection and isolation versus modelling for control. *Mathematics and Computers in Simulation*, volume 53, pages 259–271. 2000.

M. Guerfel, G. Mourot, J. Ragot, M. Benrejeb, K. Benothman. Comparaison des indices de détection de changements des modes de fonctionnement par ACP. Cas des indices basés sur l'estimation d'état. *1st International Workshop on Systems Engineering Design & Applications, SENDA'08*, Monastir, Tunisia 24-26 October, 2008.

M. F. Harkat, G. Mourot, J. Ragot. Diagnostic de fonctionnement de capteurs d'un réseau de surveillance de la qualité de l'air par analyse en composantes principales. *Journal européen des systèmes automatisés*, volume 39. pages 417–436, 2005.

M. F. Harkat, G. Mourot, J. Ragot. An improved PCA scheme for sensor FDI: Application to an air quality monitoring network. *Journal of Process Control*, volume 16, pages 625–634, 2006.

Y. Huang, J. Gertler, T. McAvoy. Sensor and actuator fault isolation by structured partial PCA with nonlinear extensions. *Journal of Process Control*, volume 10, pages 444-459, 2000.

H.P. Huang, C.C. Li, J.C. Jeng. Multiple multiplicative fault diagnosis for dynamic processes via parameter similarity measures *Industrial and Engineering Chemistry Research*, volume 46, pages 4517-4530, 2007.

I. T. Jolliffe. *Principal component analysis.* Springer-Verlag, New York, 2003.

M. Kano, H. Ohno, S. Hasebe, I. Hashimoto. New multivariate statistical process monitoring method using principal component analysis. *Computers and Chemical Engineering*, volume 25, pages 1103–1113, 2001.

M. Kano, K. Nagao, S. Hasebe, I. Hashimoto, H. Ohno, Strauss R. Comparison of multivariate statistical process control monitoring methods with applications to the Eastman challenge problem. *Computers and Chemical Engineering*, volume 26, pages 161–174, 2002.

W. Ku, R. H. Storer, C. Georgakis. Disturbance detection and isolation by dynamic principal component analysis. *Chemometrics and Intelligent Laboratory Systems*, volume 30, pages 179–196, 1995.

W. E. Larimore. Canonical variate analysis in identification, filtering and adaptative control. *29th Conference on Decision and Control*, Honolulu, Hawai Islands December, pp. 635–639, 1990.

C. Lee, S. W. Choi, I. B. Lee. Sensor fault identification based on time-lagged PCA in dynamic processes. *Chemometrics and Intelligent Laboratory Systems*, volume 70, pages 165–178, 2004.

W. Li, S. J. Qin. Consistent dynamic PCA based on errors-in-variables subspace identification. *Journal of Process Control*, volume 11, pages 661–678, 2003.

J. Mina, C. Verde. Fault detection for large scale systems using dynamic principal components analysis with adaptation. *International Journal of Computers, Communications & Control*, volume 2, pages 185–194, 2007.

S. J. Qin, R. Dunia. Determining the number of principal components for best reconstruction. *Journal of Process Control*, volume 10, pages 245–250, 2000.

S. J. Qin. Statistical process monitoring: basics and beyond. *Journal of Chemometrics*, volume 17, pages 480–501, 2003.

R. Sharmin, S. L. Shah, U. Sundararaj. A PCA Based Fault Detection Scheme for an Industrial High Pressure Polyethylene Reactor. *Macromolecular Reaction. Engineering*, volume 2, pages 12–30, 2008.

A. Singhal, D. E. Seborg. Clustering multivariate time-series data. *Journal of chemometrics*, volume 19, pages 427–438, 2005.

A. Simoglou, E. B. Martin, A.J. Morris. Statistical performance monitoring of dynamic multivariate process using state space modelling. *Computers and Chemical Engineering*, volume 26, pages 909–920, 2002.

J. Rissanen. Modelling by shortest data description. *Automatica*, volume 14, pages 465–471, 1978.

M. Tamura, S. Tsujita. A study on the number of principal components and sensitivity of fault detection using PCA. *Computers and Chemical Engineering*, volume 31, pages 1035–1046, 2007.

R. J. Treasure, U. Kruger, J. E. Cooper. Dynamic multivariate statistical process control using subspace identification. *Journal of Process Control*, volume 11, 661–678, 2004.

S. Valle, L. Weihua, S. J. Qin. Selection of the number of principal components : The variance of the reconstruction error criterion with a comparison to other methods. *Industrial and Engineering Chemistry Research*, volume 38, pages 4389–4401, 1999.

V. Venkatsubramanian, R. Rengaswamy, K. Yin, S. N. Kavuri. A review of fault detection and diagnosis; part I: quantitative model-based methods. *Computers and Chemical Engineering*, volume 27, pages 293–311, 2003.

M. Wax, I. Ziskind. Detection of the number of coherent signals by the MDL principle. *IEEE Transactions on Acoustic, Speech and signal processing*, volume 37, pages 1190–1195, 1989.

L. Xie, J. Zhang, S. Wang. Investigation of Dynamic Multivariate Chemical Process Monitoring. *Chinese Journal of Chemical Engineering*, volume 14, pages 559–568, 2006.

S. J. Zhao, J. Zhang, Y. M. Xu. Monitoring of processes with multiple operating modes through multiple principle component analysis models. *Industrial Engineering Chemical Research*, volume 43, pages 7025–7035, 2004.

LoopRank: A Novel Tool to Evaluate Loop Connectivity

M. Farenzena and J. O. Trierweiler

Group of Integration, Modelling, Simulation, Control and Optimization of Processes (GIMSCOP)
Department of Chemical Engineering, Federal University of Rio Grande do Sul (UFRGS)
Rua Luiz Englert, s/n CEP: 90.040-040 - Porto Alegre - RS - BRAZIL,
Fax: +55 51 3308 3277, Phone: +55 51 3308 4163
E-MAIL: {farenz, jorge}@enq.ufrgs.br

Abstract: Since the number of loops in refineries or petrochemical plants is very large and the number of loops with poor performance is equally large, to prioritize their maintenance is essential to ensure plant profitability. This work proposes a methodology called *LoopRank* to compute the importance factor of each loop, aiming to prioritize their maintenance. The algorithm is based on the connection among them, which is computed using partial correlation. The algorithm is based on *PageRank*, which analyses connections among nodes recursively and computes a rank for each node using partial correlation. The *LoopRank* assigns an individual score for each loop ranging from 0% to 100%. Based on this score, the loop maintenance can be ranked. The *LoopRank* algorithm is computationally efficient, thus allowing its industrial large-scale application. The proposed algorithm was applied both on simulation and industrial case studies, providing fruitful results.

Keywords: Performance Monitoring, Data correlation, Loop Rank, Partial Correlation, Data-mining.

1. INTRODUCTION

Nowadays, it is a common knowledge the positive impact of control loop performance assessment tools over industrial plants. In the last twenty years, many methodologies and tools have been developed to diagnose the main loops problems:

- Poor performance (Harris, 1989, Huang *et al.*, 1997, Jelali, 2006);
- Plant-wide disturbances (Jiang *et al.*, 2007, Thornhill and Horch, 2007, Xia *et al.*, 2005);
- Valve hysteresis (Choudhury *et al.*, 2004, Hagglund, 2002, Hagglund, 2007, Rossi and Scali, 2005, Ruel, 2000);

It is also well known that most of industrial loops do not perform well (Paulonis and Cox, 2003). However, improve and maintain all loops in their optimal performance are impossible and economically infeasible because of the small number of engineers responsible to maintain a large number of loops. Therefore, a methodology to prioritize loop maintenance is required.

Methodologies to prioritize loop maintenance or to evaluate loop interaction are scarce in the literature. Tangirala et al. (2005) proposed a method based on spectral correlation between loops. Thornhill et al. (2002) proposed tools based on spectral principal component analysis.

The scope of this work is to provide an importance score for each control loop to prioritize its maintenance. Fig. 1 shows one simple case with four loops.

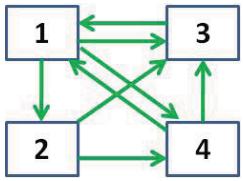


Fig. 1: Case study with four control loops interconnected.

In Fig. 1 scheme, it is easy to see that loop 3 has the most connections from others (3). So, is it the most important loop? On the other hand, loop 1 receives 2 connections, where one of them is very important, coming from loop 3, and it is the only loop 3 connection. Which is more important, loop 1 or 3? It is clear that an algorithm to systematize this procedure that provides an importance score for each loop is strongly required.

In this work is proposed an algorithm, called *LoopRank* that provides a grade based on loops connections. The loops that receive more connections from others, i.e. the loops that have stronger correlation with the remaining should have more importance than a loop that does not have any correlation with the others. To quantify these bounds, partial correlation is used. Subsequently, the priority of each loop is ranked using *PageRank* algorithm (Bryan and Leise, 2006).

The paper is segmented as follows: in section 2 the necessary background will be summarized. In section 3, the methodology to prioritize loop maintenance, proposed in this work is described. In section 4, the methodology is applied in simulation and industrial case studies. The paper ends with the concluding remarks.

2. BACKGROUND

This section provides the necessary background to understand the proposed methodology described in section 3.

2.1 Correlation and Partial Correlation

Correlation can be described as the linear dependence between two random variables (Bilodeau and Brenner, 1999). The correlation (ρ_{XY}) between two variables can be computed as follows:

$$\rho_{XY} = \frac{cov(X,Y)}{\sqrt{var(X)var(Y)}} \quad (1)$$

Where $cov(X,Y)$ is the covariance between X and Y and $var\,var$ is the variance. The correlation is a measurement of variable interaction, independently of the scale which it is measured.

In the case where the inputs and outputs are correlated, a better measure of the interaction is the partial correlation. It provides the degree of association of X and Y, with the effect of a set controlling variables (Z) removed. The partial correlation between X and Y with Z fixed ($\rho_{XY|Z}$) is computed by:

$$\rho_{XY|Z} = \frac{\rho_{XY} - \rho_{XZ}\rho_{YZ}}{\sqrt{(1-\rho_{XZ}^2)(1-\rho_{YZ}^2)}} \quad (2)$$

2.2 Importance score

To rank the relative importance of elements is essential when resources are limited. Rank algorithms have a broad class of applications including financial decisions, searching tools, among others. One rank algorithm that has been highlighted recently is *PageRank* (Bryan and Leise, 2006), which is used by the Google®'s search engine to rank pages relevance. It gives an importance score for each webpage according to an eigenvector of a weighted link matrix. It is based on the links made to a given page from other pages, and the relative impact of each source page.

The algorithm can be summarized as follows. Suppose n elements where the relative connectivity of them (x_k) should be computed, where k is the indexing element ($1 \leq k \leq n$), where this value corresponds to the arrows in each element. In the example (Fig. 1) $x_1=2$, $x_2=1$, $x_3=3$, $x_4=2$. Thus, loops can be ranked as follows 3, 1 and 4, and 2, based only on the connections. Following, the relative importance of k (x_k) is computed using the number of back links for this page. If page j contains n_j links to other pages, and one of them links to element k, then it will be boosted by a score x_j/n_j. Let $L_k \subset \{1,2,\dots,n\}$ denote the set of pages with link to page k. The relative weight for each k is computed by:

$$x_k = \sum_{j \in L_k} \frac{x_j}{n_j} \quad (3)$$

It is also assumed that the link from a page to itself has zero weight. For loop 1, its impact can be written as $x_1 = \frac{x_3}{1} + \frac{x_4}{2}$, since pages 3 and 4 have back-links to 1 and loops 3 and 4 have 1 and 2 links, respectively.

This linear relation can be written as $Ax = x$, where A is called "link matrix" and $x = [x_1 \;\; x_2 \;\; x_3 \;\; x_4]$. $A(i,j)$ provides the relative weight from loop j to loop i, where the rows show the relative weight of each connection that goes to a given loop and columns show the relative importance of the connections that come from the same loop.

In the scheme of Fig. 1, the A matrix can be written as:

$$A = \begin{bmatrix} 0 & 0 & 1 & \frac{1}{2} \\ \frac{1}{3} & 0 & 0 & 0 \\ \frac{1}{3} & \frac{1}{2} & 0 & \frac{1}{2} \\ \frac{1}{3} & \frac{1}{2} & 0 & 0 \end{bmatrix} \quad (4)$$

In the case of loop 1 (Fig 1): three arrows come out this loop. Then the relative importance added in each loop is $\frac{1}{3}$, as shown in the first column of A.

This procedure transforms the problem into a simple eigenvector problem ($Ax = \lambda x$). It can be proved that λ has always a unitary eigenvalue for this kind of matrices. Thus, the eigenvector x with eigenvalue 1 for matrix A is seek. The importance score (*IS*) for each page is given by the mentioned eigenvector just normalizing each elements by the sum of all components so that at the end the final sum is equal to 1. For the case study, the already normalized importance score are $x_1=0.387$, $x_2=0.129$, $x_3=0.290$, $x_4=0.194$.

More information about *PageRank* algorithm can be found in Bryan and Leise (2006).

3. METHODOLOGY DESCRIPTION

This section describes the *LoopRank* algorithm to evaluate loop importance.

Initially the loop output data is collected. Only routine data is required and no further information about the loop is required.

The first step is to compute the links between loops and its relative weight, where the impact of a single variable over each other should be computed. The measure of loops connectivity used in this work is the linear dependence among them.

Industrial loops generally have high correlation between them. To overcome this constraint and isolate the individual loop impact over each other, partial correlation is used. Simple correlation between loops has also been tested and results were poorer. This comparison will be shown in the case studies section. Thus, the relative weight between loops i

and j is provided by the partial correlation between these loops, removing the effect of the remaining loops ($\rho_{ij|LI}$), where $L = \{1,2,\dots,n\}$, $LA = \{i,j\}$, and $LI = L \cap LA$. Each element of the relative weight matrix (A_{ij}) is given by the partial correlation between loop i and loop $_j$ ($\rho_{ij|LI}$) :

$$A_{ij} = \rho_{ij|LI} \tag{5}$$

The next step is to evaluate *the LoopRank (LR)*, based on *relative weight matrix* (*A*), using *PageRank* algorithm that can quantify the relative importance of each loop, allowing to rank the loops for maintenance purposes. This class of algorithm was chosen because of its capacity to prioritize elements based on the connections among them and its computational/numerical efficiency. The *LoopRank* output is then normalized to limit each grade between 0 and 100%, where always the worst important loop has $LR = 0\%$ and the most important $LR = 100\%$.

Some loops can have more impact in plant profitability or help to smooth the operation. The loops that have connections with these "important ones" should have stronger weights. Thus, it is necessary to assign a *loop weight* (w_k), which is dependent on the source loop. The w_k is assigned heuristically, depending on loop type and its profitability. One heuristics is here suggested: flow and level loops are least important ($w=1$), pressure loops have middle importance ($w=1.5$) and temperature and composition are the highest importance ($w=2$). The *connection weight* k is then multiplied by all links where the source is loop k, in A.

The application of *LoopRank* algorithm can be summarized as follows:

1. Collect routine operating data of the loops;
2. Compute the partial correlation between each loop and build the *relative weight matrix (A)*;
3. Based on *A*, compute the *LoopRank*, using *PageRank* algorithm;
4. To the result in 3, multiply each loop by the corresponding loop weight (w);
5. Normalize the final results to express the result in relative percentage, where each importance is bounded between 0% and 100%.

4. CASE STUDIES

This section shows the application of *LoopRank* algorithm both on simulation and industrial case studies.

4.1 Simulation example I

In the first case study, a set of 10 loops will be analyzed. The first one has oscillatory behaviour. The other loops have been generated from the first loop by a simple addition of different noise levels followed by normalization in the amplitude. Fig. 2 shows the time trends for all loops.

Applying the proposed algorithm with $w = 1$ for all loops produces the results shown in Tab. 1. As it was already expected, the results of Tab. 1 clearly indicates that the source of oscillatory behaviour is related to the most important loop, which is in this case is the first one.

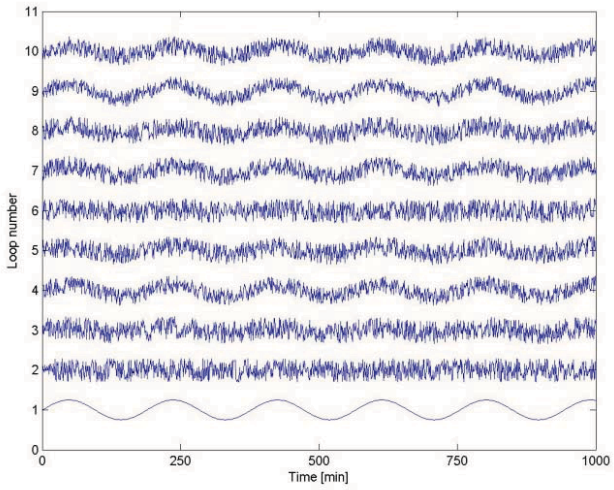


Fig. 2: Time trends of 10 data series.

Tab. 1: *LoopRanks* for case study 1.

k	LR_k	k	LR_k
1	100	**6**	16
2	3	**7**	10
3	1	**8**	20
4	21	**9**	11
5	13	**10**	0

Following, the impact of w_k will be analyzed. In this case study, w_5 will be increased and its impact over loop 4 (LR_4) and loop 8 (LR_8) are shown in Tab. 2. Since interaction between loop 5 and 1 is stronger than all remaining loop 5 connections, it is expected that increase the weight of loop 5, the *LR* of loop 1 will increase while the *LR* of all others will decrease, because the connections between loop 5 and all others will become weaker.

Tab. 2: Impact of w_5 in LR_4 and LR_8 *LoopRanks.*

w_5	LR_4	LR_8
1	21.0	20.0
1.5	20.0	18.6
2	19.3	17.3
5	16.1	12.3
10	13.7	8.3

The previous claim is corroborated by Tab. 2, where increasing w_5, the importance factors LR_4 and LR_8 decreased. LR_1 remains for all cases equal to 100%.

4.2 Simulation example II

In the second case study a set of 100 loops are be analyzed. The time trend for each loop is generated using the following procedure:

- Loop 1, loop 2, and loop 4 have oscillatory behaviour with different frequencies;
- Loop 3 and loop 5 data trends are obtained passing white-noise through a first order transfer function with different time constants;
- Loops 6, 7, and 8 are random signals;
- Other 92 data trends are generated by the linear combination of the first 8 data trends. Following, white noise is added in each one of the 92 data trends. Time trends 1, 2, and 3 impact all 92 loops using a random weight between 0.5 and 1.
- Time trends 4 to 8 impact some of 92 loops using a random weight between 0.5 and 1. The probability of each time trend to impact each loop is 50%.

Fig. 3 shows the 8 time trends for the source loops and Fig. 4 shows loops 9 to loop 18 time trends, generated using the previous loops.

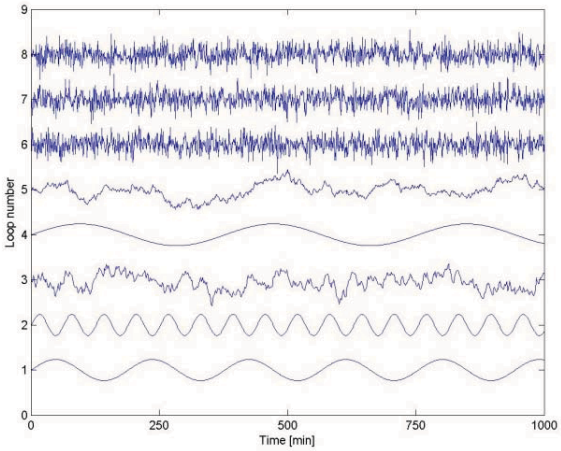


Fig. 3: Time trends for loops 1 to 8 in case study 2.

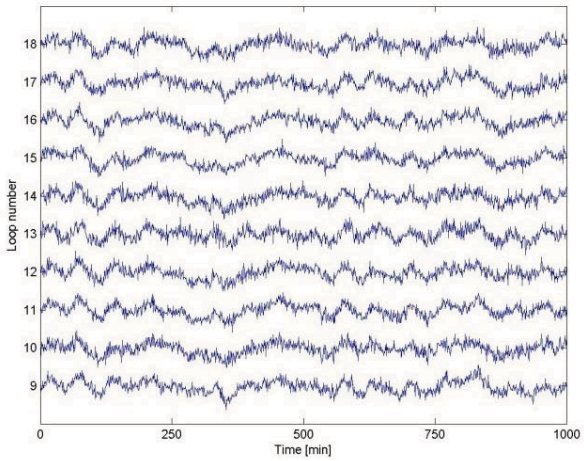


Fig. 4: Time trends for loops 9 to 18 in case study 2.

Applying the *LoopRank* algorithm, the following importance, shown in Fig. 5, is computed:

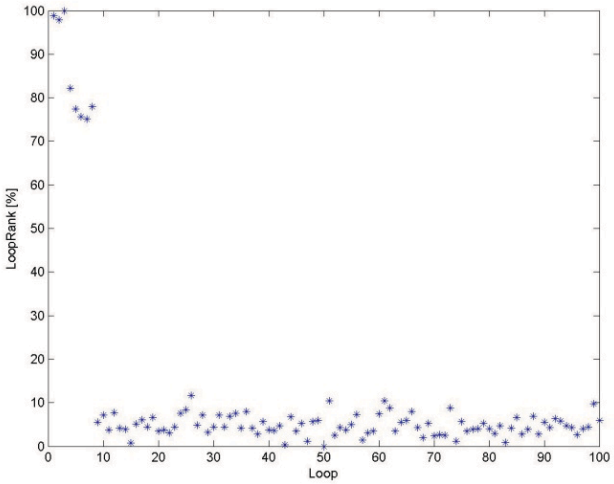


Fig. 5: *LoopRank* for case study 2, using 100 loops.

Fig. 5 reflects the expected result – loops 1, 2, and 3 have the highest importance, because of their impact in all loops. Loops 4 to 8 are less important than loops 1 to 3, but they are more important than the remaining. The remaining loops are less important, because the impact of a single one is not transferred to others.

One question can arise: If instead of partial correlation the correlation would be used, the results would be different? The comparison between *LoopRanks* using partial correlation and correlation is shown in Fig. 6.

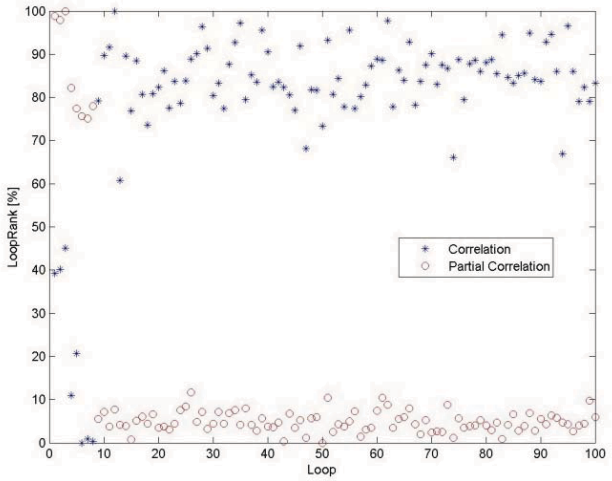


Fig. 6: *LoopRank* for case study 2, using both correlation and partial correlation.

Fig. 6 shows contrasting results, while correlation shows small importance of loops 1-8, partial correlation showed that these loops are the most important. Similar results have been seen in all tests, where correlation cannot point out the loops with major interaction among them, reason why partial correlation is used.

4.3 Industrial data

An industrial data set was provided by the courtesy of a Brazilian refinery. The system is an atmospheric distillation column of a petroleum refinery. The provided data set consists of 25 process variables: 6 level, 12 flow, 5 pressure, and 2 temperature controllers. The whole dataset has 1000 samples with a sampling time of 1 min. Fig. 7 shows the time trends of the variables.

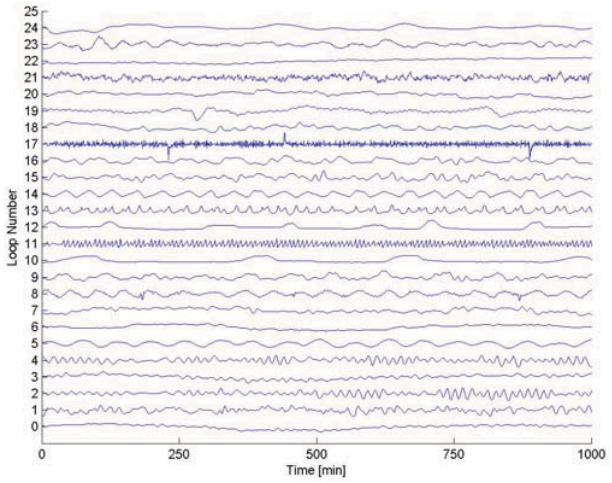


Fig. 7: *Time trends of industrial case study with 19 process variables.*

The *LoopRank* algorithm is then applied, providing the ranking shown in Fig. 8.

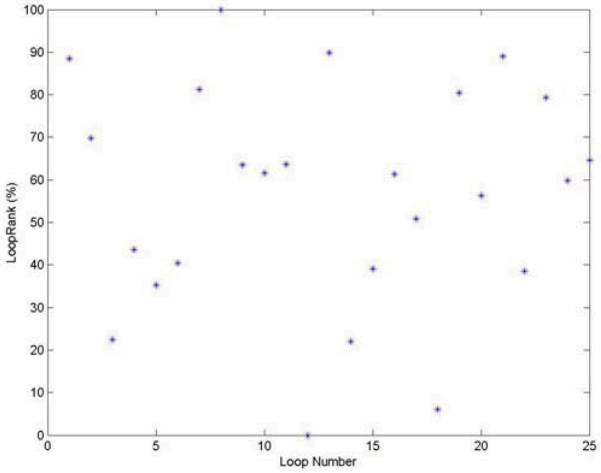


Fig. 8: *LoopRank* for industrial case study.

Fig. 8 clearly ranks the loop importance. It shows that loops 1, 8, 13, and 21 are the most important, because of its impact over the others. Loops 3, 12, 14, and 18 are the least important.

There previous results can be explained by the positions in the process flow diagram. The most important loop, in this case, is the flow of the intermediate recycle in the atmospheric tower (loop 8). The remaining most important loops are:

- Loop 1: crude oil inlet flow;
- Loop 13: Total reflux flow;
- Loop 21:Kerosene flow side-withdraw.

5. CONCLUSIONS

The main conclusions of the proposed work can be summarized as:

Loop ranking is an important tool for loop maintenance – Loop ranking for maintenance is required because of the large number of loops with poor performance in process plants. Unfortunately, the number of methodologies to evaluate loop impact is scarce. In this work a methodology for loop ranking aiming their maintenances, called *LoopRank*, is proposed.

It is better use partial correlation than correlation – *LoopRank* is based on loop interaction, measured by partial correlation. Correlation was also tested, however the results were poorer, therefore should not be used.

The proposed algorithm is similar to the Pagerank algorithm used by Google search engine – the relative importance score for each loop is computed using the *PageRank* algorithm. When the impact of a given loop should be emphasized, a loop weight can be assigned.

Successful applications of the LoopRank algorithm – the proposed algorithm was applied in 3 case studies, where reliable results were provided. One industrial case study was

presented to demonstrate the efficacy of the algorithm. The computational time for all case studies was negligible.

ACKNOWLEDGMENTS

The authors would like to thank CAPES / Brazil for supporting this work.

REFERENCES

Bilodeau, M. and Brenner, D. (1999) *Theory of multivariate statistics,* New York, Springer.

Bryan, K. and Leise, T. (2006) The $25,000,000,000 eigenvector: The linear algebra behind Google. *SIAM Review,* 48**,** 569-581.

Choudhury, M. A. A. S., Shah, S. L. and Thornhill, N. F. (2004) Diagnosis of poor control-loop performance using higher-order statistics. *Automatica,* 40**,** 1719-1728.

Hagglund, T. (2002) A friction compensator for pneumatic control valves. *Journal of Process Control,* 12**,** 897-904.

Hagglund, T. (2007) Automatic on-line estimation of backlash in control loops. *Journal of Process Control,* 17**,** 489-499.

Harris, T. J. (1989) Assessment of control loop performance. *Canadian Journal of Chemical Engineering,* 67**,** 856-861.

Huang, B., Shah, S. L. and Kwok, E. K. (1997) Good, bad or optimal? Performance assessment of multivariable processes. *Automatica,* 33**,** 1175-1182.

Jelali, M. (2006) An overview of control performance assessment technology and industrial applications. *Control Engineering Practice,* 14**,** 441-466.

Jiang, H., Shoukat Choudhury, M. A. A. and Shah, S. L. (2007) Detection and diagnosis of plant-wide oscillations from industrial data using the spectral envelope method. *Journal of Process Control,* 17**,** 143-155.

Paulonis, M. A. and Cox, J. W. (2003) A practical approach for large-scale controller performance assessment, diagnosis, and improvement. *Journal of Process Control,* 13**,** 155-168.

Rossi, M. and Scali, C. (2005) A comparison of techniques for automatic detection of stiction: simulation and application to industrial data. *Journal of Process Control,* 15**,** 505-514.

Ruel, M. (2000) Stiction: The hidden menace. *Control Magazine*.

Tangirala, A. K., Shah, S. L. and Thornhill, N. F. (2005) PSCMAP: A new tool for plant-wide oscillation detection. *Journal of Process Control,* 15**,** 931-941.

Thornhill, N. F. and Horch, A. (2007) Advances and new directions in plant-wide disturbance detection and diagnosis. *Control Engineering Practice,* 15**,** 1196-1206.

Thornhill, N. F., Shah, S. L., Huang, B. and Vishnubhotla, A. (2002) Spectral principal component analysis of dynamic process data. *Control Engineering Practice,* 10**,** 833-846.

Xia, C., Howell, J. and Thornhill, N. F. (2005) Detecting and isolating multiple plant-wide oscillations via spectral independent component analysis. *Automatica,* 41**,** 2067-2075.

Operational Flexibility of Heat Exchanger Networks

M. Escobar and J.O. Trierweiler

Group of Intensification, Modelling, Simulation, Control and Optimization of Processes (GIMSCOP), Department of Chemical Engineering, Federal University of Rio Grande do Sul (UFRGS) Porto Alegre, Brazil (*e-mail: escobar@enq.ufrgs.br/ jorge@enq.ufrgs.br*)

Abstract Process integration is motivated from economic benefits, but it also impacts on the plant behavior introducing interactions and in many cases making the process more difficult to control and operate. A prerequisite for optimal operation is that the HEN is sufficiently flexible, i.e. it must have the ability to operate over a range of conditions while satisfying performance specifications. In this work it is defined the Operational Flexibility related not only to the size of the feasible region but also to the costs involved to put the HEN into operation. In order to provide an appropriated metric, the operational flexibility index is defined. Five different networks structures designed for the nominal conditions of a case study are used to illustrate the proposed ideas. It was noticed that a great feasible region does not point out the more economic operation, and the costs must be considered together with the flexibility analysis. These characteristics are taken into account by the novel proposed operational flexibility index, which can also consider during the analysis the increasing in the utility duties, extra utility exchangers and bypass installation. These results clearly point out for the need of a simultaneous framework for flexible design and profitability.

Keywords: Heat Exchanger Networks, Optimal Operation, Operational Flexibility.

1. INTRODUCTION

Operability issues are very important for heat integrated process, since the economic performance of a process is greatly affected by process variations and the ability of the system to satisfy its operational specifications under external disturbances or inherent modelling uncertainty.

Methods based on pinch analysis and mathematical programming for fixed operating conditions hve been largely developed. An extensive review of these methods can be found in Furman and Sahinidis (2002). Compared to design of HENs for nominal operating conditions, less effort has been dedicated to the operability and controllability aspects of such networks.

Since the concept of resilient HENs firstly developed by Marselle et al. (1982) and the introduction of the flexibility index by Swaney and Grossmann (1985) several design methods based on the multiperiod approach were proposed. Floudas and Grossmann (1986) introduced a multiperiod case based on the synthesis with decomposition. Papalexandri and Pistikopoulos (1994) and Konukman et al. (2002) extended the simultaneous synthesis to the multiperiod case in an MINLP problem.

All these works relates the flexibility with the size of the feasible region and they do not take into account explicitly all the trade-offs involved in a HEN design. In this work a new metric for comparing different HEN structures is proposed based on the concept of operational flexibility. A case study with 5 different synthesized HENs is used to illustrate the proposed metric.

2. OPERATION OF HENs

A HEN is considered optimal operated if the targets temperatures are satisfied at steady state (main objective); the utility cost is minimized (secondary goal); and the dynamic behaviour is satisfactory (Glemmestad, 1997).

During HEN operation, degrees of freedom or manipulated inputs are needed for control and optimization. The different possibilities are shown in Figure 1: 1-Utility Flowrates; 2-Bypass fraction; 3-Split fraction; 4-Process Streams flowrates; 5-Exchanger area (e.g. flooded condenser); 6-Recycle (e.g. if exchanger fouling is reduced by increased flowrates).

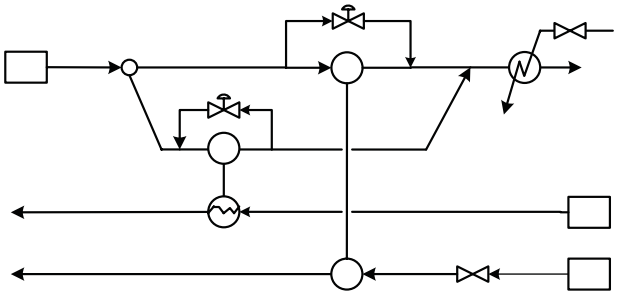


Fig. 1. Possible manipulated inputs in HENs.

In this work, we will consider the outlet target temperatures as controlled variables and utility loads, bypasses or splits, when they are present, as manipulated variables. The idea is to maintain the targets temperatures using the minimal

increase of the external utilities. The best HEN is the one where the effect of a given set of disturbances can be accommodated internally without requiring too much external "help" from the utilities heat exchangers. These ideas are illustrated through the case study of the next section.

3. CASE STUDY

To analyze the flexibility problem we have synthesized 5 different HENs for the plant illustrated in Figure 2.

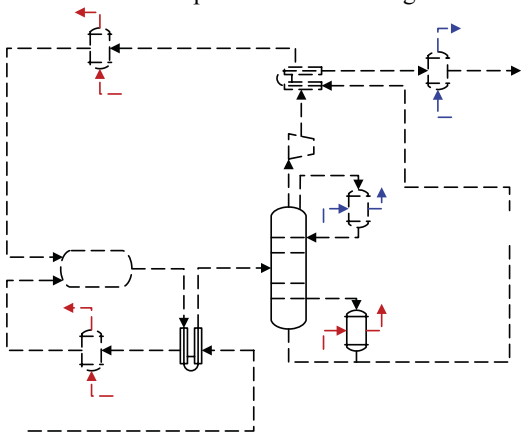


Fig. 2. Simple process with reaction, separation and heat exchangers.

Table 1. Nominal operating condition for the Case Study.

Stream	T_{in} (°C)	T_{out} (°C)	F (kW°C^{-1})	h (kW m^2 °C^{-1})
H1	270	160	18	1
H2	220	60	22	1
C1	50	210	20	1
C2	160	210	50	1
CU	15	20		1
HU	250	250		1

Cost of Heat Exchangers ($y^{-1}) = 4000+500[Area (m^2)]$^{0.83}$
Cost of Cooling Utility = 20 ($kW^{-1}y^{-1})
Cost of Heating Utility = 200 ($kW^{-1}y^{-1})

Table 1 summarizes the corresponding data of the nominal operating conditions. This data and a ΔT_{min} of 10 °C were used to design the 5 different HENs depicted in Fig. 3. The five HENs have been designed by the following approaches:

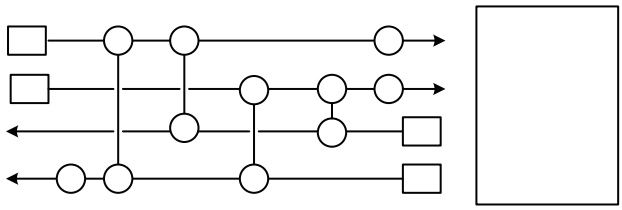
S01-Pinch Technology (Linnhoff & Hindmarsh, 1983);

S02-NLP Superstructure proposed by Floudas, Ciric, and Grossmann (1986) using Pinch Technology as initial guesses;

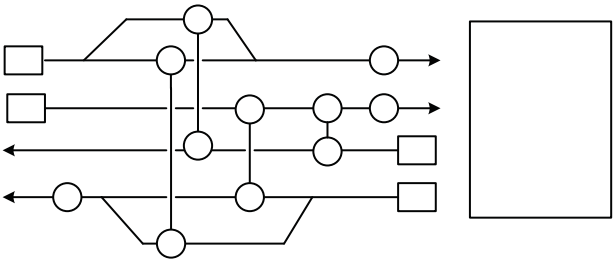
S03-NLP Superstructure in the Sequential procedure (Floudas, Ciric, and Grossmann, 1986);

S04-Hyperstrucutre proposed by Ciric and Floudas (1991); and

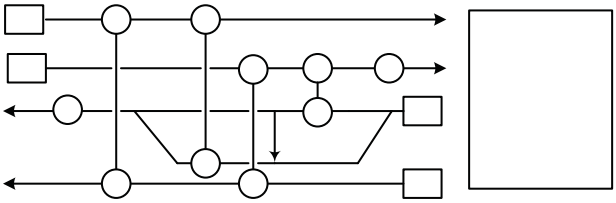
S05- the stage-wise Synheat model proposed by Yee and Grossmann (1990) with the assumption isothermal mixing.



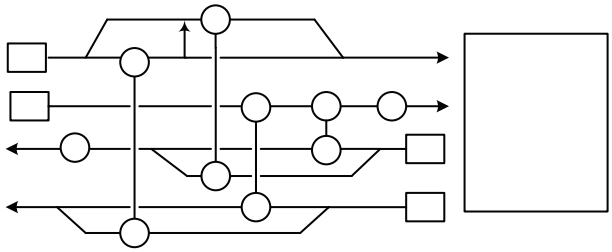
S01: Pinch Technology



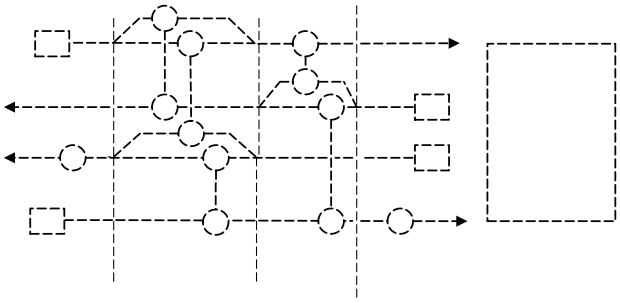
S02: NLP Superstructure (initial point by Pinch Technology)



S03: NLP Superstructure (Sequential Procedure)



S04: MINLP Hyperstrucuture (Simultaneous Procedure



S05: MINLP Synheat Model (Isothermal Mixing)

Fig. 3. Synthesized HENs for the Case Study using different approaches.

4. OPERATIONAL FLEXIBILITY

The flexibility is defined by Swaney and Grossmann (1985) as the size of the region of feasible operation in the space of possible deviations of the parameters from their nominal values. In order to analyze the flexibility, a disturbance scenario is explored on the basis of the vertices of the polyhedral region of uncertainty (Konukman et al., 2002) trough a scalar δ (flexibility target). For a fixed HEN topology and design the 'flexibility index' is defined by Swaney and Grossmann (1985) as the maximum scalar δ* (Figure 4).

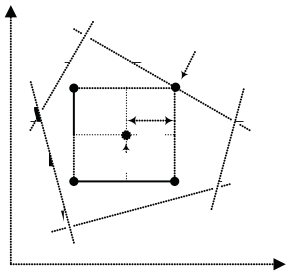


Fig. 4. Geometric representation of vertex-based flexibility target.

As the feasible region is convex when it is considered the inlet temperatures as uncertain parameters, the critical point that limits the operation lies at a vertex of the polyhedral region of uncertainty. For non-convex region the vertex-based formulation should be replaced by a more general active-constraint-strategy-based on MINLP formulation (Floudas, 1995).

Considering the four inlet temperatures as disturbances, a total of 16 vertices are enumerated. Each vertex represents an operating condition and it is formed by a deviation of ±δ from the nominal values In order to calculate the expected variations in the operating conditions that potentially could happen for a given flexibility target, each HEN configuration was implemented in Excel® using the heat exchanger model described by the set of equations (1), (2), and (3) and notation shown in Fig. 5.

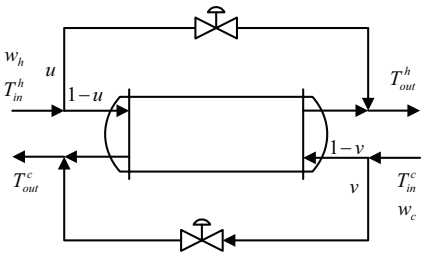


Fig. 5. General structure of a heat exchanger with bypasses.

$$T_{out}^h = (1-u)\left[\frac{(R_h-1)}{R_h-a}T_{in}^h + \frac{(1-a)}{R_h-a}T_{in}^c\right] + uT_{in}^h \quad (1)$$

$$T_{out}^c = (1-v)\left[\frac{R_h(1-a)}{R_h-a}T_{in}^h + \frac{a(R_h-1)}{R_h-a}T_{in}^c\right] + vT_{in}^c \quad (2)$$

Where

$$R_h = \frac{w_h(1-u)}{w_c(1-v)}; NTU_h = \frac{UA}{w_h(1-u)}; a = e^{NTU_h(1-R_h)} \quad (3)$$

The individual heat exchanger model was connected according to the topology for each HEN structure and the outlet temperatures deviations from their target values are calculated together with the additional utility requirement. A free simulation for fixed bypass and split fractions was carried out for each operating condition. Positive values encountered of heat duties at the stream where no utility exchanger exist mean that an extra utility exchanger must be included. Moreover, the negative values indicate an infeasible operation without any structural modifications, even for adding a new utility exchanger.

4.1 Optimal Operation of HENs

To overcome an infeasible operation it is possible to use the degrees of freedom, such as split fractions and bypasses placement in order to increase the feasible region and ensure that the optimal operation can be achieved by minimizing the utility consumption. The optimal steady-state operation or network optimization problem (Marselle et al., 1982):

Optimal Steady State Operation: *(For each operating point n)*

Minimum Utility Consumption ***(secondary objective)***

$$\min_{u,v} \sum_{i=1}^{NH} Q_{i,n}^{CU} + \sum_{j=1}^{NC} Q_{j,n}^{HU}$$

subject to.

Hot and Cold target temperatures ***(primary goal)***

$$T_{i,n}^{out} - T_i^{sp} = 0; T_{j,n}^{out} - T_j^{sp} = 0$$

Positives or zero heat loads coolers and heaters

$$T_i^{sp} - T_{i-1,n}^{out} \le 0; T_{j,n}^{out} - T_j^{sp} = 0$$

Hot and Cold Utility loads

$$Q_{i,n}^{HU} = w_i^H\left(T_{i-1,n}^{out} - T_{i,n}^{out}\right)$$

$$Q_{j,n}^{CU} = w_j^C\left(T_{j,n}^{out} - T_{j-1,n}^{out}\right)$$

Heat Exchanger Static Model

(3), (4), and (5)

*Topology Constraints**

Bypass bounds

$$0 \le u, v \le 1$$

** The topology constraints define the configuration, and are expressed as appropriated model variables connections.*

The optimal optimization problem for each configuration was implemented using the software GAMS and solved using the solver CONOPT considering δ_T is equal to 10°C (flexibility target). The new requirements for the each HEN structure are exhibited in Table 2.

According to the initial analysis, the maximum or critical utility exchanger operation is not a good metric since it was not able to distinguish the configurations S01 and S02. Furthermore, comparing the configurations S03 and S04, even though the critical loads are greater for the first one the total heat load (summation for each operation point) and the averages are not.

Table 2. Utility loads (kW) for a feasible operation for each case study using extra utility units.

Struc.	Utility	Maximum	Average	Total
S01	cold	1000	446	7584
	hot	1300	646	10984
S02	cold	1000	445	7563
	hot	1300	645	10963
S03	cold	1300	570	9886
	hot	1480	769	13073
S04	cold	1287	586	9966
	hot	1466	782	13306
S05	cold	1000	497	8455
	hot	1480	696	11840

According to the results the configurations S03 and S04 are the worst from a flexibility point of view, since they require more utility to a feasible operation. On the other hand, S04 is the HEN with lowest TAC (3.619 x10^5 \$/year) as shown in Fig. 3, but considering the flexibility this is not the best option and clearly points out that flexibility issues must be considered in an early stage of the process design, since the nominal optimum .

4.2 Optimal Operation with no extra utility units

The solution provided in the previous analysis is trivial and may guarantee the operation for a large range. Furthermore, it is an expensive solution. Providing a more reasonable analysis, a second optimal operation problem was considered. The new problem definition differs from the previous one by the addition of constraints that ensure no extra utility exchangers. The general results are presented in Table 3.

Table 3. Utility loads (kW) for a feasible operation for each case study using no extra utility units.

Struc.	Utility	Maximum	Average	Total
S01	cold	1000	494	8425
	hot	1714	694	11825
S02	cold	1003	502	8534
	hot	1540	702	11934
S03*(8)	cold	1058	521	8851
	hot	1480	721	12251
S04*(14)	cold	902	499	8410
	hot	1480	699	11810
S05*(7)	cold	1000	530	9011
	hot	1587	730	12411

* (*ni*) indicate *ni* infeasible operating points.

Due to extra constraints, greater utility consumption in general was need. Moreover, how it was expected not always a feasible solution could be found. The main difficult faced by the configurations S03, S04 and S05 was the presence of only two utility exchangers, i.e. these configurations are more penalized with the additional constraints. The bad performance of the configuration S05 may be also explained possibly by the "inflexible" isothermal mixing constrain applied to the design.

A new analysis was made considering the possibility of variation for the extra degrees of freedom, when they take place. Whereas the configuration S01 has no one split fraction, the best possible results has already presented in Table 3. Conversely, all other configurations have split fractions. For the configurations S03 and S04 was also considered as an extra degree of freedom the recycle stream, from the outlet of a heat exchanger to another. The results are presented in Table 4.

Table 4. Utility loads (kW) for a feasible operation for each case study using no extra utility units but using extra degrees of freedom (split and recycle fractions).

Struc.	Utility	Maximum	Average	Total
S01	cold	1000	494	8425
	hot	1714	694	11825
S02	cold	900	435	7396
	hot	1430	635	10796
S03*(7)	cold	990	458	7780
	hot	1383	658	11180
S04*(8)	cold	780	417	7088
	hot	1368	641	10904
S05	cold	900	456	7745
	hot	1431	656	11145

* (*ni*) indicate *ni* infeasible operating points.

Like it was expected the extra degrees may be used to achieve the targets and decreases the utility consumption increasing the feasible region, which is proven by the increase of the number of feasible operating points. For the configuration S05, allowing the manipulation of the split fractions automatically removes the isothermal mixing assumption and hence increases considerably the flexibility.

Comparing the results, the configurations S03 and S04 must be discarded because they do not provide a suitable operation. The results are a sign of designs with splits are good from the flexibility viewpoint because these extra degrees of freedom can be used to decrease the investment cost during the design phase and be used to decrease the utility consumption during operation. In addition, the installed areas are utilized completely for all operating points, which not occurs using bypasses. In the overall design the dynamic behaviour must be analysed carefully once split fractions can give competitive effects.

4.3 Flexibility Range

All the previous analysis considered the flexibility target (δ_T) of 10^{o}C, in order to analyze the flexibility range, the total utility consumption (δQ) levels corresponding to the critical operating conditions versus the flexibility targets (δ_T) for structures S01, S02 and S05 and the virtual structure (Maximum Energy Recovery) MER were calculated and they are shown in Figure 6.

The illustration reveals plateaus of total utility requirements levels for a given value of δ, under the correspondent δQ level the configuration is operable, i.e. it will not violate the temperatures specifications as long as the deviations in the

source streams temperatures along the vertex directions have magnitudes within $0 < \delta_T < \delta$.

The analysis reveals the trade-off between the flexibility target and the total utility load need to maintain a feasible operation pointing out that a more flexible is more expensive. For practical purposes, increasing the flexibility target trough penalization of total utility consumption is possible until a limit (δ*), which is reached when at least one bypass saturation occurs.

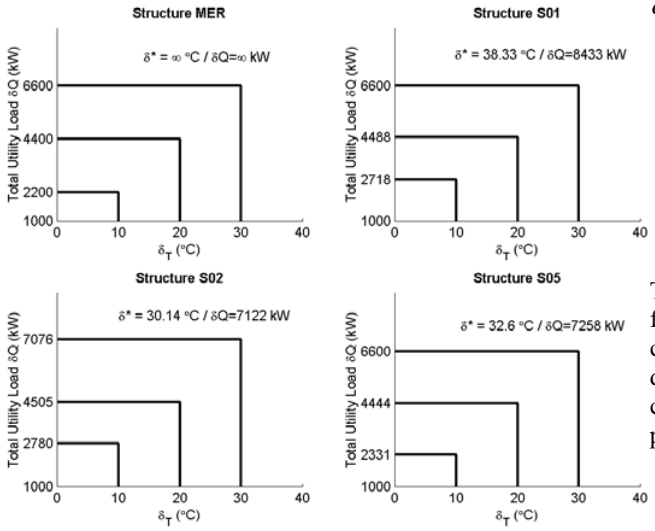


Fig. 6. Total utility consumptions at the critical operating conditions versus the flexibility ranges for structures MER, S01, S02 and S05.

In Table 4, the structure S02 ($\delta^*=38.33^0$C) depicted the lowest total utility load in general (considering all operating points) and the lowest average utility load. Therein, the critical loads define the feasibility operational range and it must be checked, but a selection of a structure using purely the analysis provided by the Figure 6 will not be appropriated because it would assume that most of the time the process would operate in the critical conditions what is not correct.

5. OPERATIONAL FLEXIBILITY INDEX

An appropriated metric to compare different HENs is based on the operational flexibility that is reached if the operation is possible and the maximum energy recovery is obtained for the entire feasible region with a minimum investment cost.

The structure of the HEN has a direct influence on the flexibility. Disintegrated structures are highly flexible, but that trivial solution is not interesting under an economic point of view. The other highly flexible possibility is a totally integrated structure, with the maximum number of units and maximum areas with bypasses across all units, but very expensive from an investment point of view.

Here we introduce the operational flexibility index to take into account in addition to the feasible range related with a flexibility target the most important costs involved during a "flexible operation". The Operational Flexibility Index for a specific flexibility target (OF_δ) is defined in equation (4), where the two terms correspond to operating cost (φ_{oc}) and the investment cost (φ_{ic}) penalties for an operational flexibility, and these terms are defined in equations (5) and (6). The operational flexibility index varies form 0 to 100%. Its upper bound indicates feasible operation without much economic penalty. On the other hand, when bypasses, new units, increased areas, and increased utility consumption are considered the indice will be penalized.

$$OF_\delta(\%) = 100(1-\varphi_{oc}-\varphi_{ic}) \tag{4}$$

$$\varphi_{OC} = w_1 \frac{1}{2(V+1)} \sum_{n=1}^{V} \frac{\sum_{k=1}^{NH+NC} \delta q_{k,n}^{U} - \sum_{k=1}^{NH+NC} \delta q_{k,n}^{U,\min}}{\sum_{k=1}^{NH+NC} \delta q_{k,n}^{U,\max} - \sum_{k=1}^{NH+NC} \delta q_{k,n}^{U,\min}} \tag{5}$$

$$\varphi_{IC} = w_2 \frac{N_{bp}}{N_{hx,retrofit}} + w_3\left(1 - \frac{N_{hx}}{N_{hx,retrofit}}\right) + w_4\left(N_{hx} - \sum_{m=1}^{N_{hx}} \frac{A_{m,installed}}{A_{m,retrofit}}\right) \tag{6}$$

The parameters w_i (7) correspond to the normalized weight for each contribution to the penalty. A suggested set may be calculated by the constants k_i (8) that depends on economic data from the process, i.e. the utility costs and the exchanger costs, considering the bypass cost. For the case study these parameters are provided in Table 1.

$$w_i = \frac{k_i}{\sum_{i=1}^{4} k_i} \tag{7}$$

$$k_1 = \frac{cu.}{2(V+1)}\left(\sum_{k=1}^{NH+NC} \delta q_{k,n}^{U,\max} - \sum_{k=1}^{NH+NC} \delta q_{k,n}^{U,\min}\right)$$

$$k_2 = 0.2aN_{hx,ret}; k_3 = aN_{hx,ret}; k_4 = b\sum_{m=1}^{N_{hx}} A_{m,ret}^{\beta} \tag{8}$$

The parameters N_{bp}, N_{hx} and A_m correspond to the number of bypasses placed, the number of heat exchangers and the area of the heat exchanger m, respectively. Moreover, the subscript 'retrofit' indicates the variable in the flexible operation, i.e. the retrofitted design.

To evaluate the potential of each structure, the operational flexibility index was calculated. The calculation requires the bounds for the utility loads. It was used the LP transshipment model (Papoulias and Grossmann, 1983) for each operating point to estimate the minimum utility consumption for the design case ($\Delta T_{min}=10^0$C) and the minimum case ($\Delta T_{min}=0^0$C). Furthermore, it was calculated the utility loads for the no heat integration case; all the targets are exhibited in Table 5.

Table 5. Utility loads (kW) for a feasible operation for each case study using no extra utility units.

Case	Utility	Max.	Average	Total
MER	*cold*	900	412	7000
$\Delta T_{min}=10^oC$	*hot*	1300	612	10400
MER	*cold*	900	219	3720
$\Delta T_{min}=0^oC$	*hot*	1080	360	6120
No Heat	*cold*	5900	5500	93500
Integration	*hot*	6400	5700	96900

The main results are expressed in Table 6. The term corresponding to the energy cost is dominant due to its greater economic impact in the total cost; the investment cost is worthless for most cases. The structure S02 showed the best performance for the required flexibility target. The interpretation inside the context of a feasible operation is that a greater index indicates that operation occurs inside a more economic way, using a lower average utility consumption with the lower investment cost. Otherwise different conditions will penalize the operational flexibility.

Table 6. Operational Flexibility Index for the structures S01, S02 and S05.

	S01	*S02*	*S05*
φ_{oc}	0.0605691	0.0498255	0.0536638
φ_{ic}	0.00203939	0.00271918	0.00000000
$OF_{\delta=10}$	93.73916%	94.74553%	94.63362%

5.1 Flexibility x Installed Area

All the previous analysis was carried out using the areas as fixed parameters, and these areas were designed at nominal conditions. If it was considered the whole feasible region, trough a multi-period design these areas would have better usage in order to reduce the utility consumption in the entire region. A new optimization problem was performed for the structure S01, considering varying areas. In order to avoid extreme solutions, a practical consideration for the areas bounded between 1 and 1000 m^2 were imposed and new optimizations were performed. The areas for each operating point are presented in Table 7. In order to satisfy all operating points, the maximum areas obtained in Table 7 where fixed and the optimal operation problem was solved with the increased areas.

Table 7. Nominal and maximum areas (m^2) for the HEN structure S01.

	$A_{H1,C1}$	$A_{H1,C2}$	$A_{H2,C1}$	$A_{H2,C2}$
Nominal	318.12	56.55	609.97	209.79
Maximum	1000.00	97.46	1000.00	1000.00

Comparing the values obtained with the results presented in Table 4 for the structure S01, the total utility loads decreased from 8425.3kW to 4370.9 kW (cold utility) and 11825.3 kW to 7770.9 kW (hot utility); and the average consumption decreased from 494kW to 257kW(cold utility) and 694kW to 457kW (hot utility). The flexibility index (δ*) provided in the Figure 6 increased from 38.33^{0}C to 49.8^{0}C, i.e. the feasible region increased. Furthermore, the operational flexibility index (OF_δ) exhibited in Table 6 increased from 93.73916% to 98.197858% considering only the energy cost and considering the capital cost for the oversize of the areas the index is 97.02954%.

6. CONCLUSIONS

The flexibility analysis of different structures previously designed was accomplished through optimal operation problem taking into account the trade-offs between energy cost, capital cost and the flexibility in order to ensure an economic operation. The formulation presumed that the feasible region in the space of uncertain input parameters was convex, and thus the optimal solution was explored based on the vertices of the polyhedral uncertainty region in the space of source-stream temperatures. It was defined the operational flexibility index as a measure of operational flexibility that was assumed to be different of structural flexibility. The first one considers the impacts on the total annual cost, since infinity areas, high levels of utility loads and disintegrated structures are according with this work highly structural flexible but present a poor (expensive) operation and hence a low operational flexibility.

The HEN structure provides an upper bound for the flexibility that should be expected during operation. The increasing of flexibility target reveals the flexibility dependent on structural modifications and total utility consumption until the unfeasible operation may be achieved. It was showed that more important that the size of the feasible region it is the cost involved in a feasible operation around the desired flexibility target. It has shown the real need of taking into account the flexibility in a simultaneous framework, once the utility loads, heat exchangers (units and areas), and the arrange (configuration of flows, temperatures) are determined in only one step, and all these variables strongly affect the flexibility.

REFERENCES

Ciric, A. R. & Floudas, C. A. (1991). Heat exchanger network synthesis without decomposition. Computer and Chemical Engineering15, 385.

Floudas, C. A., Ciric, A. R. & Grossmann, I. E. (1986). Automatic synthesis of optimum heat exchangers network configurations. American Institute of Chemical Engineering Journal 32, 276.

Floudas, C.A., Grossmann, I.E., (1986). Synthesis of flexible heat exchanger networks for multiperiod operation. Computers & Chemical Engineering 10 (2), 153–168.

Furman K.C. and N.V. Sahinidis,(2002). A critical review and annotated bibliography for heat exchanger network synthesis in the 20th century, Industrial and Engineering Chemistry Research 41 (10), pp. 2335–2370.

Glemmestad, B. (1997). Optimal Operation of Integrated Processes, Studies on Heat Recovery Systems. Ph.D. thesis, Norwegian University of Science and.

Konukman A.E.S., M.C. Camurdan, U. Akman, (2002). Simultaneous flexibility targeting and synthesis of minimum-utility heat exchanger networks with superstructure-based MILP formulation, Chem. Eng. Processing 41 501–518.

Linnhoff B., E. Hindmarsh, (1983) The pinch design method for heat exchanger networks, Chem. Eng. Sci. 38 (5) 745–763.

Marselle, D.F., Morari, M., Rudd, D.F., (1982). Design of resilient processing plants—II, design and control of energy management systems. Chemical Engineering Science 37 (2), 259–270.

Papalexandri, K.P., Pistikopoulos, E.N., (1994). Synthesis and retrofit design of operable heat exchanger networks—I, flexibility and structural controllability aspects. Industrial & Engineering Chemistry Research 33 (7), 1718–1737.

Papoulias, S. A., & Grossmann, I. E. (1983). A structural optimization approach in process synthesis-I. utility systems. Computers and Chemical Engineering, 7, 695–706.

Swaney R.E., and I.E. Grossmann (1985). An index for operational flexibility in chemical process design. AIChE J. 31 (4) (1985) 621–630.

Yee, T. F., & Grossmann, I. E. (1990). Optimization models for heat integration-II. Heat exchanger network synthesis. Computers and Chemical Engineering, 14, 1165–1184.

GPC Controller Performance Monitoring and Diagnosis Applied to a Diesel Hydrotreating Reactor

A. C. Carelli* M. B. da Souza Jr.**

*Chemical Engineering Department, Federal University of Rio de Janeiro, Rio de Janeiro Brazil (e-mails: *alain.carelli@gpi.ufrj.br, **mbsj@eq.ufrj.br)*

Abstract: Control systems tend to lose performance over time if their responses are not monitored and thus there is no support information on to how to make adjustments on them. Reliable controllers have complementary systems to identify and diagnose reductions in performance and also to implement predetermined solutions vis-à-vis the desirable type of output. The goal of this work was to analyze controller performance monitoring and causes diagnosis methods based in two indexes: historical benchmark and model based performance measurement. These methods were applied to situations of degraded performance simulated in the predictive control of a hydrotreating reactor, aiming the identification of the reduction in the controller performance and the discrimination of its causes. The obtained results can also be extended to several other chemical processes, once that the investigated process presents first order with dead-time dynamics, typical of these processes.

Keywords: 1. Process control. 2. Performance reduction detection and diagnosis. 3. Control audit. 4. Refinery.

1. INTRODUCTION

In recent years, the performance requirements for process plants have become increasingly difficult to satisfy. Stronger competition, tougher environmental and safety regulations, and changing economic conditions have been key factors in tightening product quality specifications. A further complication is that modern plants have become more difficult to operate because of their complex and highly integrated processes. The largest emphasis recently given to safety has naturally improved the importance of the process control area. Without process control systems integrated with computers, it would be impossible to operate modern plants safely and lucratively while achieving product quality and environmental requirements. Therefore, it becomes important for chemical engineers to have an understanding both of the theory and of the process control practice. (Seborg, Edgar and Mellichamp, 2004).

Controller performance assessment and monitoring are necessary in order to assure the process control effectiveness and profit of the plant. The initial design of control systems includes many uncertainties caused by approximations in process model, estimations of disturbance dynamics and magnitudes, and assumptions about operating conditions. Many factors can cause their abrupt or gradual performance deterioration overtime. Around 60% of all industrial controllers have some kind of performance problem (Schäfer and Cinar, 2004).

All controllers need to be retuned as the dynamic of the process suffer natural or continuous alterations. The controllers performance should be monitored, because, even though they may have been adequately adjusted, it is expected that their performance decays along years due to variations in the materials, deterioration of the instrumentation, changes in the plant, etc. This reduction in the performance should also be diagnosed, enabling the identification of the needs to readjust the controller tuning parameters.

The main benefit of applying advanced control strategy to catalytic processes in refineries can be related to quality giveaway. For hydrotreating units, quality giveaway is mainly obtained by reducing over-desulphurization. Experimental results showed clearly that the sulfur content of the product is strongly related to the severity of the reaction, which is determined by reactor bed temperatures and the residence time. Operating at higher temperatures yields better product quality, but at the same time shortens the catalyst cycle. Therefore, the better the reaction control is (so as to guarantee only the necessary conversion), the better the utilization of the catalyst cycle and the lower the operational cost of the process (Lababidi, Alatiqi and Ali, 2004). Additionally, in case of accident, the replacement of a reactor and the reconstruction of other damaged equipments can take up to 12 months and the cost of lost production can exceed US$ 50 millions (Ancheyta and Speight, 2007).

2. CONCEPTUAL ASPECTS

This work is structured under three main themes: dynamic process control, process control performance assessment and diesel hydrotreating.

2.1 Model Predictive Control – Generalized Predictive Control

The general set of the available Generalized Predictive Control (GPC) algorithms cover a large variety of control goals in contrast to other methods, so that some of them can even be considered GPC specific cases.

In the SISO case (single-input *u*, single output *y*), a linearized time-invariant discrete process is assumed, where the relations between input and output are described by the following equation:

$$A(q^{-1})y(t) = q^{-d}B(q^{-1})u(t-1) \quad (1)$$

A and *B* are polynomials in the backward shift operator q^{-1} with, respectively, degrees *m* and *n*, and *d* is the dead-time.

With the premise that all process natural disturbances can be characterized by a stochastic disturbance, the principle of the superposition can be used to represent all disturbances as a unique influence in the output. Then, the process can be described by the following CARIMA (controlled auto-regressive and integrated moving average) model:

$$A(q^{-1})y(t) = q^{-d}B(q^{-1})u(t-1) + C(q^{-1})e(t)/\Delta \quad (2)$$

where *C* is also a polynomial in the backward shift operator q^{-1}, *e(t)* is an uncorrelated random sequence and $\Delta(q^{-1})$ is the differencing operator $1 - q^{-1}$.

The CARIMA model may be considered the most appropriated model for many industrial applications with non-stationary disturbances. In practice, it has two main types of disturbance: occurrence of random steps in random intervals (e.g. changing of the product quality) and Brownian motion which is met in plants that depend on the energy balance (Clarke, 1988).

The following Diophantine Equation is employed for the development of the solution:

$$C(q^{-1}) = E(q^{-1})A(q^{-1})\Delta + q^{-d} * F(q^{-1}) \quad (3)$$

where, *E* and *F* are polynomials in the backward shift operator q^{-1} with degrees *d-1* and *m*, respectively.

Multiplying the term $E(q^{-1})\Delta q^{j}$ in the components of (2); considering $C(q^{-1}) = 1$ (alternatively *C* is truncated and absorbed inside the polynomials *A* and *B*); and, assuming the future error values equal to zero, because they do not depend on the past values of *y(t)* and *u(t)*, the following equation is obtained:

$$y(t+j) = G(q^{-1})\Delta u(t+j-d-1) + F(q^{-1})y(t) \quad (4)$$

where, $G(q^{-1}) = E(q^{-1})B(q^{-1})$.

In the GPC, the predictions *y(t+j)* are estimated in order to compare them with a reference trajectory, and to calculate the optimum control actions. The system outputs will be influenced by signals in *u(t)* after of the sampling periods *d+1*, due to the system dead-time of *d* sampling periods.

The following cost function is assumed:

$$J = (Gu + f - w)^T (Gu + f - w) + \lambda u^T u \quad (5)$$

where, *w* is the reference trajectory or *set-point* and λ is a weighting sequence.

Assuming that there are no constraints in the control signals, the minimum of *J* can be met by equating the *J* gradient to zero. Therefore, the following result is used in order to obtain the future control actions:

$$\Delta u = \left(G^T G + \lambda I\right)^{-1} G^T (w - f) \quad (6)$$

2.2 Predictive Control Performance Monitoring and Diagnosing

In order to perform performance reduction diagnosis of the controller, its performance shall initially be monitored preferable on-line. There is a set of techniques conceived for this purpose, named controller performance monitoring (CPM) techniques.

The objective of the CPM is to develop and implement technologies that provide information of the plant to determine if the appropriated performance and the characteristics of behavior are being reached through the controlled variables. For the case SISO, the normalized performance index is an elegant method, which compares the theoretic absolute lower limit in the output variability with the achieved values. This index could configure itself as a benchmark appropriated to measure the performance of a feedback control system (Cinar, Palazoglu and Kayihan, 2007).

Nevertheless, mostly for multivariable MPC controllers, other CPMs methods have been studied based in the calculation of the cost function, which in most cases is the objective function minimized to determine the MPC's strategy. Cinar, Palazoglu and Kayihan (2007) introduced two methods based on monitoring of the cost function values for the controller performance reduction diagnosis, called of historical benchmark and model-based performance measurement.

The cost function J_{ach} is obtained with plant real values that can be described in the following form:

$$J_{ach} = \frac{1}{Pc}\left\{\sum_{j=1}^{Pc}\left[e^T(k+j-Pc)Qe(k+j-Pc) + \Delta u^T(k+j-Pc)R\Delta u(k+j-Pc)\right]\right\} \quad (7)$$

Where, *Pc* is the moving horizon of past data; *e(k)* is the vector of control errors at time *k* (difference between the controlled variable and the reference trajectory); *Δu* is the change in manipulated variables at time *k*; and, *Q* and *R* are weighting matrices representing the relative importance of each controlled and manipulated variable.

The historical benchmark requires a priori knowledge of good performance during a certain time period according to some expert assessment. The cost function applied in historical benchmark has the same form of (7), where the input and output data are taken from that period. So, the value

achieved through this function is constant until a better performance is reached (Schäfer and Cinar, 2004).

The historical benchmark index is described by the following expression, which supports the control performance reduction or increase detection:

$$\gamma_{his}(t) = J_{his} / J_{ach}(t) \quad (8)$$

The model-based performance measure index compares the achieved performance with the performance in the design case that is characterized by inputs and outputs given by the model (Schäfer and Cinar, 2004).

The model-based performance measure index is described by the following expression:

$$\gamma_{des}(t) = J_{des}(t) / J_{ach}(t) \quad (9)$$

Both cost functions used in the calculation of the model-based performance measure index have the same form of (7).

Monitoring the model-based performance measure is useful in diagnosing causes that affect the design case controller. Two groups of causes may be devised. For instance, increases in unmeasured disturbances, actuator faults, or increase in the model mismatch do not influence the design case performance (group II causes). Accordingly, J_{des} remains constant while J_{ach} increases, reducing the model-based performance measure. Root cause problems such as input saturation or increase in measured disturbance, on the other hand, affect the design case performance as well (group I causes). This lead to an approximately constant value of the model-based performance measure, if the effect is quantitatively equal (Cinar, Palazoglu and Kayihan, 2007).

This diagnostic sequence assumes that only one source cause occurs. If γ_{des} doesn't change significantly, while the model performance and achieved performance decrease quantitatively equal, the diagnosis of the root cause is in the group I. If γ_{des} presents a considerable decrease, the diagnosis of the root cause is in the group II. In the case that multiple causes can occur simultaneously, the diagnosis logic becomes more complex.

Subgroups are defined to further distinguish between the root cause problems in group I. All changes in the controller (e.g. tuning parameters, estimator, constraints) are assumed to be performed manually. These changes are known and their effects can be monitored. However, the action taken is known and the root cause of the effect does not need to be identified by diagnosis tools (subgroup Ia causes). The remaining two root cause problems (change in measured disturbances and input saturation) belong to subgroup Ib. Additional information is needed to distinguish between the two root cause problems in subgroup Ib. Looking at the manipulated variables, input saturation can be determined by visual inspection. A saturation effect in a manipulated variable indicates input saturation as underlying root cause and rules out the increase in measured disturbances (Cinar, Palazoglu and Kayihan, 2007).

2.3 Diesel Hydrotreating

The hydrotrating unit considered in this work is the Trickle Bed Reactor (TBR) with two reactors in series, each reactor formed by two fixed bed, as showed in Figure 1.

The oil feed is combined with makeup hydrogen and recycle hydrogen and heated to the reactor inlet temperature. Heat is provided from heat exchange with the reactor effluent and by a furnace. The reaction of hydrogen and oil occurs in the reactors in the presence of the catalyst. To prevent reactor temperatures from getting too high, quench gas (cold recycled hydrogen gas) is added between reactors and between catalyst beds of multiple-bed reactors to maintain reactor temperatures in the desired range.

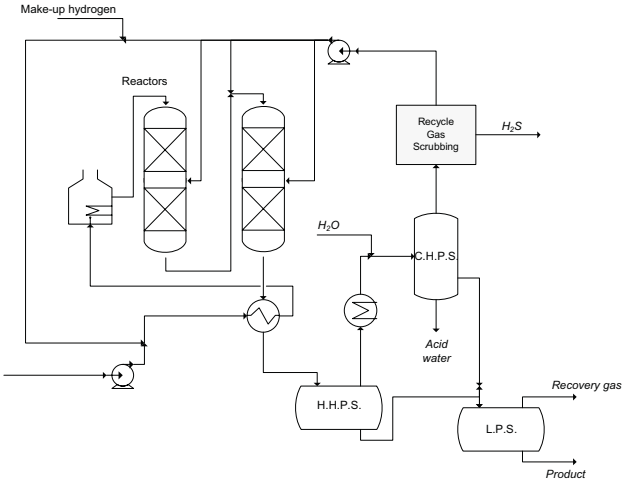


Figure 1 – Diesel hydrotreating process

The second reactor effluent is cooled (by exchange with the reactor feed) to recover the heat released from the hydrotreating reactions. After cooling, the reactor effluent is flashed in the hot, high-pressure separator (HHPS) to recover hydrogen and to make a rough split between light and heavy reaction products. The liquid from HHPS has its pressure lowered, than it is sent to the low-pressure separators, and on to the product fractionator. The HHPS vapor is cooled and water is injected to absorb hydrogen sulfide and ammonia produced in the reactors by the hydrotreating reactors. The mixture is further cooled to condense the product naphtha and gas oil and is flashed in the cold, high-pressure separator (CHPS). The CHPS separates the vapor, liquid water, and the liquid light hydrocarbons. The pressure of the hydrocarbon liquid is lowered and it is sent to the low-pressure separators. The water is sent to a sour water recovery unit for removal of the hydrogen sulfide and ammonia. The hydrogen-rich gas from the CHPS flows to the H_2S absorber. The purified gas flows to the recycle compressor where it is increased in pressure so that it can be used as quench gas and recombined with the feed oil. Liquid from the low-pressure separators is fed to the atmospheric fractionator, which splits the hydroprocessed oil from the reactors into the desired final products.

The model adopted in this work to represent the HDT's process was the model presented by Carneiro (1992) which applies the concept proposed by Hlavácek (1982) in representing fixed beds through the CSTR-CELL model. The CSTR-CELL in series describes the adiabatic fixed bed reactors dynamic. In Figure 2, a scheme of this model is shown.

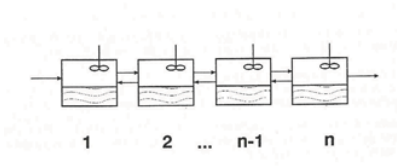


Figure 2 – CSTR-CELL reactor model

The CSTR-CELL reactor model considers mass and heat axial dispersion in the bed, mass diffusion and heat transportation between fluid and solid phases, as illustrated in Figure 3. The following assumptions are adopted in the CSTR-CELL: only one first order reaction – with respect to the mean concentration of a pseudo-reagent "A" in the solid phase porous – occurs and the reaction rate can be described by the Arrhenius equation; there is no volume variation in the reactor; the reactors are adiabatic; there is only one liquid and one solid phase with constant physical-chemical properties; there is only longitudinal transport phenomena; and, there are non-linear interactions between kinetic and thermal processes.

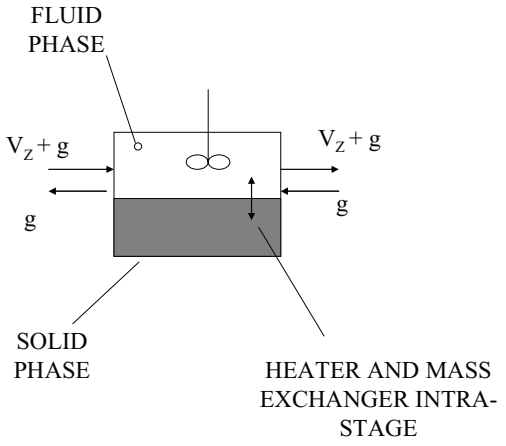


Figure 3 – CSTR-CELL stages

The Carneiro (1992) model was employed in this work for being at the same time able to represent the main process dynamics and simple, as it is composed by ordinary differential equations.

4. METHODOLOGY

This paper focus on the primary controller of the cascade control system applied to the first bed of the first reactor of the HDT unit, which can be seen in the top left hand corner of the diagram shown in Figure 4. This controller controls the bed outlet temperature through the manipulation of the set-point that is sent to the secondary controller. The secondary controller controls the inlet temperature of the bed through the manipulation of the fuel flow that enters the furnace.

The primary controller was performed by the GPC algorithm, using no explicit constraints and weighting in the cost function. The tuning parameters were the prediction horizon (N), the control horizon (NU) and the reference trajectory parameter (α).

The GPC was projected with a first order internal model with dead-time. The function considered for reference trajectory was a first order equation, which has only one tuning parameter: α. The larger α, the more cautious the control actions. If α is zero, the trajectory is constant and equal to the set-point, as can be noticed in the equation to follow:

$$w(t+1) = \alpha\, y(t) + (1-\alpha) SP \tag{10}$$

As a default option, α was chosen in this study as equal to 0.7.

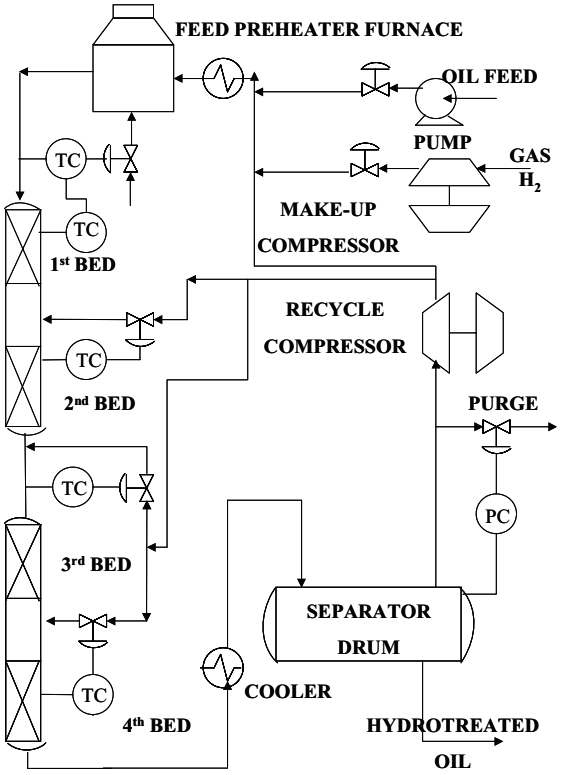


Figure 4 – Diesel Hydrotreating Unit Diagram (De Souza Jr., Campos and Tunala, 2009)

In respect to the assumed performance reduction scenarios, four cause diagnosis cases were tested based in the method presented previously: increased controlled variable variability (group II causes), mismatch between the model and the phenomenological model based simulator (group II causes), saturation of manipulated variable (group Ib causes) and change of the control tuning parameters (group Ia causes).

5. RESULTS

The events responsible for the reductions in performance were introduced in the 80th sampling time, as can be observed in the following figures. Figures 5, 6, 7 and 8 present, respectively, the following situations: increase in the controlled variable variability, mismatch between the internal model of the controller and the phenomenological model based simulator, saturation of the manipulated variable and change of the control tuning parameter. In all figures, the first graph presents the cost functions calculated to obtain the historical benchmark and model-based performance measure indexes. The second graph presents the historical benchmark index (monitoring index), and the third graph presents the model-based performance measure index (diagnosis index).

With the controlled variable variability increased in 5 times (in the phenomenological simulator), the achieved cost function was increased, while the others remained at the same level, as shown in Figure 5. In consequence, both the monitoring and diagnosis indexes had their values reduced. So, these behaviors agree with the expected for causes belonging to the group II which is the case for increases in unmeasured disturbances.

Figure 6 represents the change of CARIMA model parameters – *a* and *b* of (2) – which were multiplied by 20, causing an increase of the achieved cost function. The cost function applied to the model remained at the same level, because the internal model of the controller was affected in the same way.

As the mismatch between the internal model of the controller and the phenomenological simulator belongs to the causes of group II, the monitoring and diagnosis indexes decrease as can be seen in Figure 6.

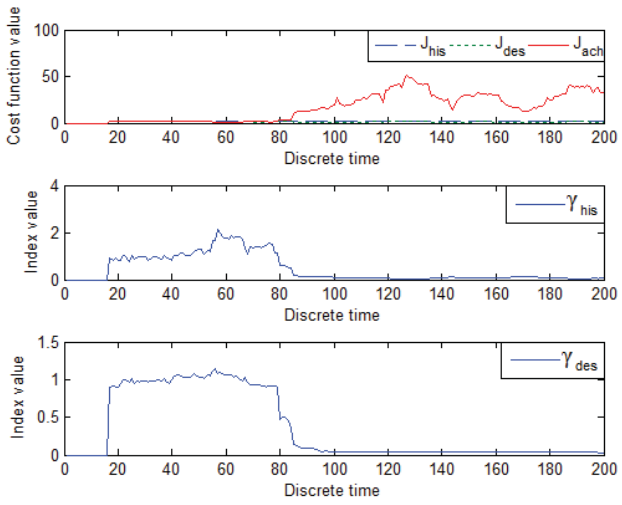


Figure 5 – Control performance reduction diagnosis caused by the controlled variable variability increase

When a constraint in the manipulated variable was applied to limit its lower value to 236.1ºC, the achieved cost function and the based-model cost function showed an increase which resulted in the decrease of the monitoring index and in the maintenance of the diagnosis index (see Figure 7). The observed behavior agrees with the causes belonging to group I, as was expected. Among the causes of group I, this particular cause can be diagnosed by monitoring the control actions, such as presented in the Figure 8, where from the 80th sampling time ahead the manipulated variable did not decrease beyond the value -0.2.

Figure 9 represents the control tuning change situation, where the prediction horizon varied from 4 to 50. In this situation, it can be observed that the indexes presented a similar behavior to the previous situation, due to the fact that this kind of cause also belongs to group I, where the same change affects the internal model and the model of the phenomenological simulator. This cause would not need to be diagnosed, because the modification in the tuning parameters of the controller – and, therefore, the reason of the performance reduction – would be previously known.

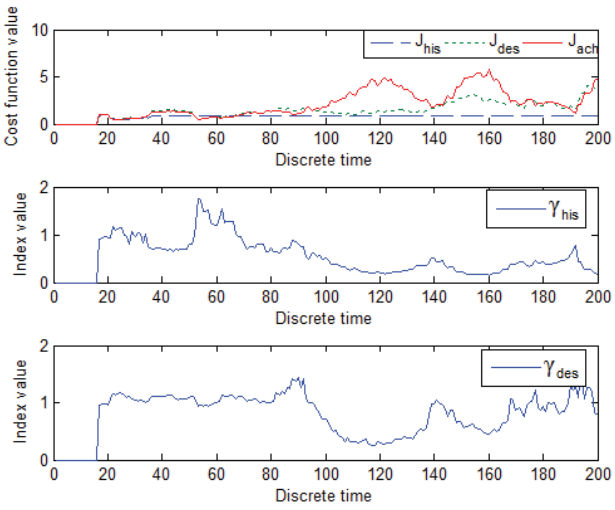


Figure 6 – Control performance reduction diagnosis caused by the mismatch between the model and the phenomenological model based simulator

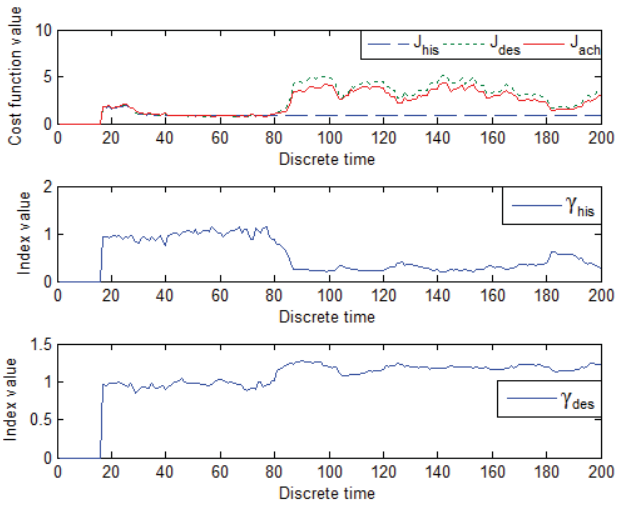


Figure 7 – Control performance reduction diagnosis caused by the saturation of the manipulated variable

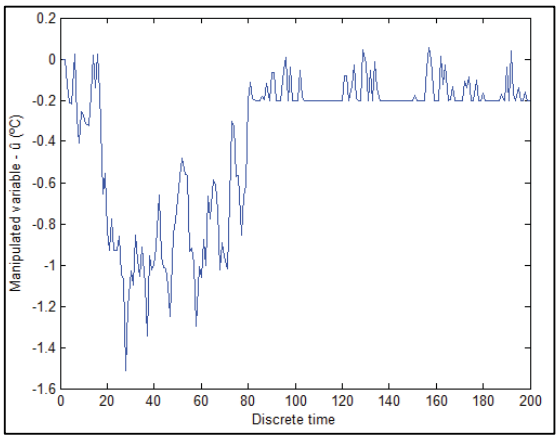


Figure 8 – Saturation of manipulated variable

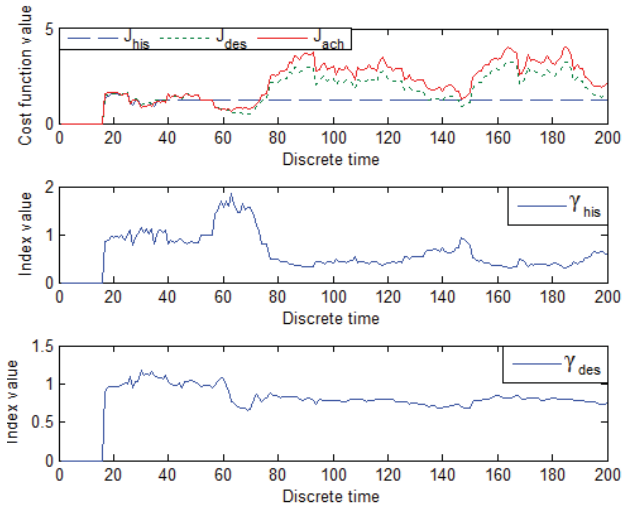


Figure 9 – Control performance reduction diagnosis caused by the change of the control tuning parameter

Even though some situations were potentially easier (e.g. the saturation of the manipulated variable) to detected and diagnose than others (e.g. the mismatch between the model and the phenomenological model based simulator), all the simulated scenarios were safely diagnosed. However, the situations studied in this paper were magnified in order to allow the verification of the differences that were expected for each case in the figures.

6. CONCLUSIONS

Monitoring and diagnosis methods were successfully applied to study control performance reduction scenarios using a GPC algorithm. Four types of performance reduction causes were diagnosed: increase in the controlled variable variability, mismatch between the model and the phenomenological simulator, saturation of manipulated variable and change in the control tuning parameter.

As future developments, it is suggested the implementation of operation support tools that enable the automatic performance monitoring and diagnosis.

Finally, it is expected that – with environmental concern, development of the industrial safety area and evolution of human intellectual capacity – more and more technologies will be developed in order to enable the correction, the prevention and, mostly, the failures prediction, allowing the man to dedicate his work to nobler activities, like process optimization.

REFERENCES

Ancheyta, J., Speight, J. G. (2007). *Hydroprocessing of heavy oils and residua*. CRC Press. USA.

Carneiro, H. P. (1992). Robust control of a fixed bed chemical reactor. 1992. 142 p. Dissertation (Chemical Engineering Master) – Engineering Pos-Graduation Program Coordination (COPPE), Federal University of Rio de Janeiro, Rio de Janeiro.

Cinar, A., Palazoglu, A., and Kayihan, F. (2007). *Chemical process performance evaluation*. CRC Press. USA.

Clarke, D. W. (1998). Application of generalized predictive control to industrial processes. *IEEE Control Systems Magazine*. April. p. 49 – 55.

De Souza Jr., M. B., Campos, M. C. M. M., and Tunala, L. F. (2009). Dynamic Principal Component Analysis Applied to the Monitoring of a Diesel Hydrotreating Unit. In Ferrarini, L. and Veber, C. (ed.). *Modeling, control and diagnosis of complex industrial and energy systems*. ISA, USA.

Hlavácek, V. (1982). *Fixed bed reactors, flow and chemical reaction*. Verlag Chemic. Germany.

Lababidi, H. M. S., Alatiqi, I. M., and Ali, Y. I. (2004). Constrained model predictive control for a pilot hydrotreating plant. *Chemical Engineering Research and Design*, v. 82 (A10), p. 1293 – 1304.

Schäfer, J. and Cinar, A. (2004). Multivariable MPC system performance assessment, monitoring and diagnosis. *Journal of Process Control*, v. 14, p. 113 – 129.

Seborg, D. E., Edgar, T. F., and Mellichamp, D. A. (2004). *Process dynamic and control*. Wiley, 2nd Edition. USA.

ACKNOWLEDGMENTS

This work was sponsored by 'Brazilian Research and Projects Financing Agency' (FINEP) and PETROBRAS (Grant 01.04.0902.00). Professor M. B. de Souza Jr acknowledges CNPq for a research fellowship.

Early determination of toxicant concentration in water supply using MHE

F. Ibrahim [*] **B. Huang** [*] **J. Xing** [**] **B. Jayasankar** [*]

[*] *Department of Chemical and Material Engineering, University of Albert, Canada T6G 2G6. fadi.ibrahim@ualberta.ca, biao.huang@ualberta.ca, jayasank@ualberta.ca*
[**] *Department of Laboratory Medicine and Pathology, University of Alberta, Canada T6G 2S2. jzxing@ualberta.ca*

Abstract: In this paper, a novel application of state estimation in environmental engineering is presented. Filtering techniques including moving horizon estimator (MHE) and extended Kalman filter (EKF) are used for early concentration estimation of toxic agents existing in water supply. The purpose is to integrate the filtering techniques with an early warning system enabling an early detection of the presence of toxicants in the water supply system and quantifying their concentrations. The estimation is based on dynamic measurements generated by a real-time cell electronic sensor (**RT-CES**) and cytotoxicity dynamic models.

Keywords: state estimation, extended Kalman filter, moving horizon state estimation, cytotoxicity, real-time cell electronic sensor, early warning, water protection.

1. INTRODUCTION

Drinking water may be contaminated by a range of chemical, microbial and physical hazards that could pose risks to health if they are present at high levels. Examples of chemical hazards include mercury, chromium, arsenic, etc. The sources of these toxicants differ with respect to the toxicant. Mercury for instance, occurs as a result of both natural (volcanic, forest fires and oceanic releases) and anthropogenic sources (mining, smelting and other industrial activities) in our environment as mentioned by Wang et al. (2004).

The effects of toxicants on the human cells are referred to as cytotoxicity. In other words, cytotoxicity is the characteristic of being toxic to living cells, including cell killing, cell lysis and certain cellular pathological changes, such as cellular morphological change and adhesion change as reported in Xing et al. (2005). Therefore, citizens must be alerted as early as possible when water is contaminated. For this purpose, an early warning system is necessary for detection of any sudden deterioration in the quality of water supply. An efficient detection must include the ability of an early determination of the presence of a toxicant at low concentration. Thus, our main objective in this paper is to use filtering techniques, such as moving horizon estimation (MHE) and extended Kalman filter (EKF) to determine on-line the concentration of such a toxicant in water supply. To achieve this purpose, mathematical modeling and real-time measurements are necessary. Two mathematical models have been developed and validated by Huang and Xing (2006) to predict cell toxicity response to mercury (II) chloride and sodium dichromate [chromium (VI)] toxicity. The measurements of toxicity response were recorded using Real-Time Cell Electronic Sensor (**RT-CES**). These two models are able to predict cell responses to different values of toxicant concentration and allow assessment of the biological consequences of toxic chemicals in environmental contamination. In this paper, we reverse the modeling procedure. We are interested in the estimation of toxicant concentration for a given dynamic model through on-line monitoring data sampled from **RT-CES**. The organization of this paper is as follows. The monitoring procedure and the mathematical models are revisited in Section 2 and Section 3 respectively. The procedure of concentration estimation and the validation results are presented in Section 4 including concentration estimation using both MHE and EKF. Concluding remarks are given in Section 5.

⋆ The authors gratefully acknowledge the support of the Natural Sciences and Engineering Research Council of Canada (NSERC).

2. EQUIPMENT AND MONITORING PROCEDURE REVISIT

1) ***Equipment***: The RT-CES system (ACEA Biosciences, CA, U.S.A.) is used for this study and has been described in (Xing et al., 2005; Huang and Xing, 2006). Briefly, as shown in Fig. 1, it consists of a 16x microelectronic sensor devices having 16 plastic wells in microtiter plate format, a device station and an electronic sensor analyzer. Cells are grown onto the surfaces of microelectronic sensors. In operation, the sensor devices with cultured cells are mounted to a device station placed inside a CO2 incubator. Electrical cables connect the device station to the sensor analyzer. Under the control of RT-CES software, the sensor analyzer automatically selects wells to be measured and continuously conducts measurements. The electronic impedance can then be transferred to a computer and recorded. A parameter termed cell index (CI) is derived to represent cell status based on the measured electrical impedance. The frequency dependent

electrode impedance (resistance) without or with cells present in the wells is represented as $R_b(f)$ and $R_{cell}(f)$, respectively. The CI is calculated by:

$$CI = \max_{i=1,\ldots,n} \left[\frac{R_{cell}(f_i)}{R_b(f_i)} - 1 \right] \quad (1)$$

where n is the number of the frequency points at which the impedance is measured. Several features of the CI can be derived: (1) Under the same physiological conditions, if more cells attach onto the electrodes, the larger impedance value leading to a larger CI value will be detected. If no cells are present on the electrodes or if the cells are not well-attached onto the electrodes, $R_{cell}(f)$ is the same as $R_b(f)$, leading to $CI = 0$; (2) A large $R_{cell}(f)$ value leads to a larger CI. Thus, CI is a quantitative measure of the number of cells attached to the sensors; (3) For the same number of cells attached to the sensors, changes in cell status, such as morphological change, lead to change of CI.

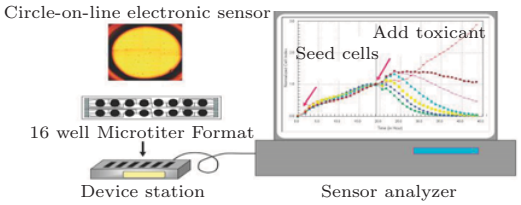


Fig. 1. *The Real-Time Cell Electronic Sensor*

In addition to cell numbers, the impedance also depends on the extent to which cells attach to the electrodes. For example, if cells spread, there will be a greater cell/electrode contact area, resulting in larger impedance. Thus, the cell biological status including cell viability, cell number, cell morphology and cell adhesion will all affect the measurements of electrode impedance that is reflected by CI on the RT-CES system. Therefore, a dynamic pattern of a given CI curve can indicate sophisticated physiological and pathological responses of the living cells to a given toxic compound (Xing et al., 2005).

2) ***Dynamic growth with toxicity***: Two environmental toxicants, mercury (II) chloride and sodium dichromate [chromium (VI)], were used for cytotoxicity assessment on the 16 sensor device. The cell line NIH 3T3 was tested. The starting cell number was 10 000 cells per sensor wells. The cell growth on the sensor device was monitored every hour up to 24 h in real-time by the RT-CES system. When the CI values reached a range between 1.0 and 1.2, the cells were then exposed to either mercury (II) chloride, or chromium (VI) at different concentrations. Fig. 2 shows dynamic cytotoxic response to different doses of chromium (VI). Fig. 3 shows dynamic cytotoxic response to mercury (II) chloride. In both cases the cytotoxicity response is dose dependent and increasing dose leads to decreasing (CI).

3. MATHEMATICAL MODELING

As mentioned in the introduction, the cytotoxicity mechanism is complex and cell response to toxicity depends on cell type, toxicant type, toxicant concentrations and the time of exposure to the toxicant. In Huang and Xing (2006), two types of models were developed and validated

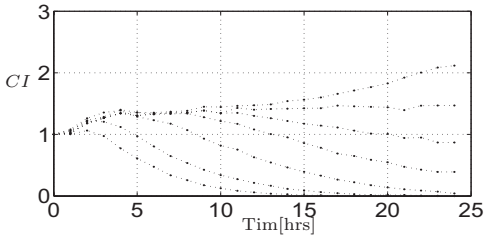


Fig. 2. *Dynamic cytotoxic response of NIH 3T3 cells to different doses of* ***chromium (VI)****: 0; 0.62; 0.91; 1.97; 2.89; 4.25; 5.78 in the unit of μM. Increasing dose leads to decreasing (CI).*

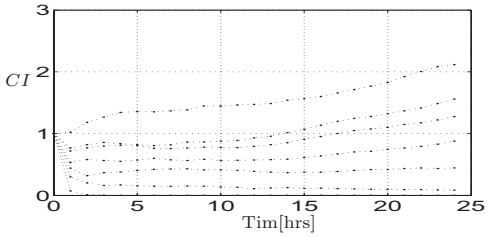


Fig. 3. *Dynamic cytotoxic response of NIH 3T3 cells to different doses of* ***mercury (II)****: 0; 10.43; 15.2; 22.35; 32.8; 48.3; 71 in the unit of μM. Increasing dose leads to decreasing (CI).*

to predict cell toxicity response to mercury (II) chloride, and sodium dichromate [chromium (VI)] stimulations. In both models, it was suggested that the process of cytotoxicity follows two-step mechanism: (1) uptake of toxicant by cells and (2) killing of the cells. The uptake mechanism describes the transport process of the toxicant into a cell as illustrated in Fig. 4 (Huang and Xing, 2006). This mechanism relates the extracellular concentration c_e (representing the concentration of a toxicant in the environment) and the intracellular concentration c_i (concentration inside the cell) and it is described by El-Kareh and Secomb (2005) as follows :

$$\dot{c_i} = k_1(k_2 c_e + \frac{k_3 c_e}{k_4 + c_e} - c_i) \quad (2)$$

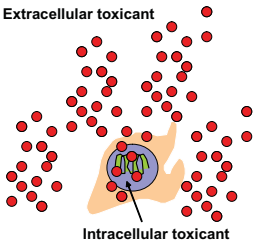


Fig. 4. *Schematic of transport process of toxicant onto cell.*

The first step in cytotoxicity (the uptake mechanism) is supposed to be rather consistent; however the second step (cell killing) differs with respect to the toxicant and can be described through cell population dynamics as $\dot{N} = f(C, N)$ where N is the cell population and C can be the intracellular or the extracellular concentration of the toxicant or a combination of them. It depends on the type of the toxicant. In the following, we present the mathematical models of cell killing under the effect of two toxicants, mercury (II) chloride, and sodium dichromate [chromium (VI)] as described in Huang and Xing (2006).

3.1 Mathematical modeling of [chromium (VI)] toxicity

The cell exposed to dichromate [chromium (VI)] is killed by *apoptosis* mechanism which is a highly regulated process and is described as programmed cell death. The apoptosis mechanism mainly depends on the intracellular concentration of the toxicant. This is described by the dynamics of cell pullulation given by Eliaz et al. (2004) as $\dot{N} = N(k_s - kc_i)$. As proposed by Huang and Xing (2006), this mechanism yields the following system of differential equations for describing dichromate [chromium (VI)] effects on cells population dynamics:

$$\begin{aligned} \dot{c_i} &= k_1(k_2 c_e + \frac{k_3 c_e}{k_4 + c_e} - c_i) \\ \dot{N} &= N(k_s - kc_i) \end{aligned} \tag{3}$$

The parameters of this model (3) are estimated from the experiment data and presented in Table 1.

Table 1. Estimated parameters for model (3)

k_1	k_2	k_3	k_4	k_s	k
0.0146	2.9399	0.0080	29.2418	0.0425	0.1041

3.2 Mathematical modeling of mercury (II) chloride toxicity

In Huang and Xing (2006), mercury cytotoxicity has both necrosis mechanism and apoptosis mechanism. The necrosis describes an accidental cell death caused, for example, by chemical or physical assault to the cell which may make cells die by direct disruption of cell membrane. Thus necrosis mechanism mainly depends on extracellular concentration of the toxicant. The cell population dynamics, together with the uptake mechanism expressed by eqn. (2), under mercury (II) chloride toxicity effect is described as follows:

$$\begin{aligned} \dot{c_i} &= k_1(k_2 c_e + \frac{k_3 c_e}{k_4 + c_e} - c_i) \\ \dot{N} &= N(k_5 + k_6 c_i + k_7 c_e) \end{aligned} \tag{4}$$

The parameters of model (4) are presented in Table 2.

Table 2. Estimated parameters for model (4)

k_1	k_2	k_3	k_4	k_5	k_6	k_7
7.735	1.108	3.21	12.8	0.0312	0.2084	-0.2364

4. RAPID TOXICANT CONCENTRATION ESTIMATION

The on-line estimation of the key parameter (the concentration of toxicant c_e) is critical due to two reasons. First, as being well known the concentration itself is usually not easily measurable due to technical or economical limitations especially for biomedical processes. The key toxicants are usually measured by high performance liquid chromatography (**HPLC**) and liquid chromatography-mass spectrometry (**LC-MS**) which are expensive equipments. Second, an early determination (detection) of the concentration of such a toxicant is important for an early warning system (which we aim to develop) in order to detect any sudden deterioration in the quality of water supply. Deterioration in water quality mainly means increase of the concentrations (or even the presence) of toxicants . For the early warning system, it is necessary to do on-line estimation of c_e.

For on-line estimation, several methods for state estimation are available such as EKF and MHE. EKF is a popular state estimation technique and considered as the standard choice for estimating state for nonlinear systems due to lower computation load and more stable property. However, additional physical insights about the process may help in state estimation to prevent negative concentration for instance. This kind of insights can not be considered in EKF. On the other hand, this physical insights can be added as inequality constraints and integrated with an optimal state estimation scheme formulated as a quadratic problem such as MHE (Rao et al., October 2001).

We use mainly MHE for on-line estimation of the key parameter c_e and we also use EKF as an alternative (usually a quicker method) of the estimation. This may be considered as a comparison to demonstrate by biological application examples the benefits of using MHE on one hand. On the other hand, the EKF is also imbedded in the MHE and is naturally used for a comparison. The superiority of MHE has also been pointed out by several authors through a number of applications such as in Rao and Rawlings (2002) and Haseltine and Rawlings (2005) for instance.

Before starting the procedure of on-line state estimation of the key parameter (the extracellular concentration c_e) for our biological application, an identifiability test is necessary. We present in the next section an identifiability test for mercury (II) chloride toxicity model (4) for an illustration. An identifiability test for chromium (VI) toxicity model (3) can be performed similarly.

4.1 Identifiability

A mathematical model is identifiable if there exist no two parameter sets which have the same input-output behavior. In other words, a model is not identifiable if there exists no unique parameter set to explain the input-output behavior. Identifiability is a pre-analysis for parameter estimation problem to determine the uniqueness of the parameter solution obtained from the estimation process. A number of methods are available for testing identifiability of parametric models. For testing the identifiability of the mercury (II) toxicity model (4), the Taylor series approach is utilized (see Walter and Pronzato (1996)). A brief description of the approach is given below. Consider the following model :

$$\begin{aligned} \dot{x}(t) &= f(x(t), u(t), t, p), \quad x(0) = x_0(p) \\ y(t, p) &= h(x(t), p) \end{aligned} \tag{5}$$

where p is the model parameters set.

If $a_k(p) = \lim_{t \longrightarrow 0+} \frac{d^k}{dt^k} y(t, p)$ then a sufficient condition for model (5) to be uniquely identifiable is :

$$a_k(\hat{p}) = a_k(p^*), \quad k = 0, 1, ..., k_{max}, \Longrightarrow \quad \hat{p} = p^*$$

where k_{max} is a positive integer, small enough for the computations to remain tractable.

Since mercury (II) toxicity model (4) has only one parameter (the concentration c_e), checking for identifiability reduces to checking for conditions under which the parameter can be observed from the Taylor series coefficients. The first two coefficients of the series for the mercury (II) toxicity model can be determined as:

$$\begin{aligned} a_0(p) &= N(0) \\ a_1(p) &= \dot{N}(0) = N(0)(k_5 + k_6 c_i(0) + k_7 c_e) \end{aligned} \tag{6}$$

Solving the equation (6) for c_e yields,

$$c_e = \frac{k_6 c_i(0) N(0) + k_5 N(0) - \dot{N}(0)}{-k_7 N(0)}$$

Therefore c_e is identifiable if:

$$-k_7 N(0) \neq 0 \quad \text{and} \tag{7}$$

$$k_6 c_i(0) N(0) + k_5 N(0) - \dot{N}(0) \neq 0 \tag{8}$$

Measurements evolution shown in Fig. 2 and values of the estimated parameters in table 2 satisfy both conditions, equations 7 and 8.

4.2 Extended Kalman Filter (EKF) formulation

Before presenting EKF formulation, considering the problem of estimating the state of system modeled by the nonlinear state space equation :

$$\begin{aligned} x_{k+1} &= f(x_k, u_k, k) + G w_k \qquad k = 0, 1, 2... \\ y_k &= g(x_k, k) + v_k \end{aligned} \tag{9}$$

where, $x_k \in \mathbb{R}^n$ is the state vector, $y_k \in \mathbb{R}^p$ the measured output, $w_k \in \mathbb{R}^n$ state disturbance and $v_k \in \mathbb{R}^p$ the measurement noise. The EKF linearizes the nonlinear system and then applies the Kalman filter to obtain the state estimation. The method can be summarized in a recursion structure similar to linear Kalman filter for a nonlinear system described above by equations (9) (see Haseltine and Rawlings (2005)):

Prediction step

$$\begin{aligned} \hat{x}_{k|k-1} &= f(\hat{x}_{k-1|k-1}, u_{k-1}, w_{k-1}) \\ P_{k|k-1} &= A_{k-1} P_{k-1|k-1} A_{k-1}^T + G_{k-1} Q_{k-1} G_{k-1}^T \end{aligned}$$

Update step

$$\begin{aligned} \hat{x}_{k|k} &= \hat{x}_{k|k-1} + K_k(y_k - g(\hat{x}_{k|k-1})) \\ K_k &= P_{k|k-1} C_k^T [C_k P_{k|k-1} C_k^T + R_k]^{-1} \\ P_{k|k} &= P_{k|k-1} - K_k C_k P_{k|k-1} \end{aligned}$$

in which, the following linearizations are made

$$A_k = \frac{\partial f(x_k, u_k, k)}{\partial x_k^T}, \quad G_k = \frac{\partial f(x_k, u_k, k)}{\partial w_k^T}, \quad C_k = \frac{\partial g(x_k)}{\partial x_k^T}$$

4.3 Moving Horizon State Estimation (MHE) formulation

The MHE strategy belongs to a class of optimization methods for on-line determination of state. The optimization problem is formulated as a least-squares problem where the decision variables are chosen to minimize the sum of the squared errors between the available measurements and model prediction. When a new measurement becomes available, this optimization is repeated by adding the new measurement to the past measurements each sampling time. This leads to growing computational burden of solving the least-squares optimization, known as full information estimation problem. MHE reduces this computational cost by considering a finite horizon of only the last N_h measurements in the optimization problem and information provided by past data beyond the horizon is captured by arrival cost. This optimization is repeated each sampling time by including a new measurement and discarding the first measurement while keeping a fixed horizon length (N_h). In other words, the MHE algorithm is a least square optimization problem solved over a window of fixed horizon length (N_h). This window moves one step ahead each time after solving an optimization problem with a quadratic cost function (Ψ_k) of the following form (for more details see Rao et al. (2003)):

$$\begin{aligned} \min_{\{\hat{w}_{k-N_h-1|k}, \ldots, \hat{w}_{k-1|k}\}} \Psi_k : \Psi_k &= \hat{w}_{k-N_h-1|k}^T Q_{-N_h|k}^{-1} \hat{w}_{k-N_h-1|k} \\ &+ \sum_{j=k-N_h}^{k-1} \hat{w}_{j|k}^T Q^{-1} \hat{w}_{j|k} + \sum_{j=k-N_h}^{k} \hat{v}_{j|k}^T R^{-1} \hat{v}_{j|k} \end{aligned}$$

subject to the state equality constrains:

$$\begin{aligned} \hat{x}_{k-N_h|k} &= \bar{x}_{k-N_h|k} + \hat{w}_{k-N_h-1|k} \\ \text{with} \quad \bar{x}_{k-N_h|k} &= f(\hat{x}^*_{k-N_h-1|k-1}, u_{k-N_h-1}, k) \\ \hat{x}_{j+1|k} &= f(\hat{x}_{j|k}, u_j) + \hat{w}_{j|k}, j = k - N_h, ..., k-1 \\ y_j &= g(\hat{x}_{j|k}, k) + \hat{v}_{j|k}, j = k - N_h, ..., k \end{aligned}$$

with the possibility to incorporate inequality constraints on the state, state disturbance and process noise:

$$\begin{aligned} w_{min} < A w_j < w_{max}, \quad x_{min} < A x_j < x_{max}, \\ v_{min} < A v_j < v_{max}, \quad j = k - N_h - 1, ..., k-1 \end{aligned}$$

where Q is the covariance of the state disturbance and R is the covariance of process noise.

The term $\hat{w}_{k-N_h-1|k}^T Q_{-N_h|k}^{-1} \hat{w}_{k-N_h-1|k}$ approximates the arrival cost which summarizes the effects of the past information before $t = k - N_h$. The weighting term $Q_{-N_h|k}$ initially represents the covariance of the prior state estimate $\bar{x}_{k-N_h}$ and is computed according to EKF covariance update formula (Rao et al., 2003):

$$\begin{aligned} Q_{-N_h|k+1} = A_k Q_{-N_h|k} A_k^T + G_k Q_k G_k^T - \\ A_k Q_{-N_h|k} C_k^T [C_k Q_{-N_h|k} C_k^T + R_k]^{-1} C_k Q_{-N_h|k} A^T \end{aligned} \tag{10}$$

where A_k, C_k and G_k result from linearizing the model (9) around the estimated trajectory.

In the full information problem there is no arrival cost because the whole information (all available measurements) is used each sampling time in the optimization while in MHE, only a subset of the information is used and the rest is approximated by the arrival cost. Thus, MHE is an approximation of the full information problem and therefore stability issue arises. The key to preserving

stability is how to approximately summarize the old data, equivalently, how to find the best approximation of the arrival cost, an explicit expression which rarely exists in nonlinear or constrained system. One strategy is to use the EKF covariance update formula as presented in equation (10) (Rao et al., 2003).

Next we present a state estimation based approach for rapid determination of concentrations of mercury (II) and chromium (VI) from the measurement of cell population responses provided by the **RT-CES**.

4.4 Concentrations estimation of chromium (VI)

The key parameter we aim to estimate is the extracellular concentration (c_e) of chromium (VI). This parameter is added to the toxicity equation (3) of chromium (VI) as an augmented state as follow :

$$\begin{aligned} \dot{c_i} &= k_1(k_2 c_e + \frac{k_3 c_e}{k_4 + c_e} - c_i) \\ \dot{N} &= N(k_s - k c_i) \\ \dot{c_e} &= 0 \quad ; \quad y = N \end{aligned} \tag{11}$$

where, c_i is the intracellular concentration, N is the cell population and y is the observation. We estimate toxicant concentrations from three toxicity responses corresponding to toxicant doses c_e =(0.62; 1.97; 4.25)μM. Note these data have not been used for modeling purpose and thus serve as cross validation data for state estimation. The results presented in Fig. (5-7) show that MHE in general has a better estimation than EKF. In addition, MHE is able to prevent an estimation of negative concentration at all time but EKF can not. It is also observed from these figures that the extracellular concentration can be correctly estimated between 10 to 15 hrs, instead of 24 hrs as traditional the method needs. Thus a rapid estimation is achieved.

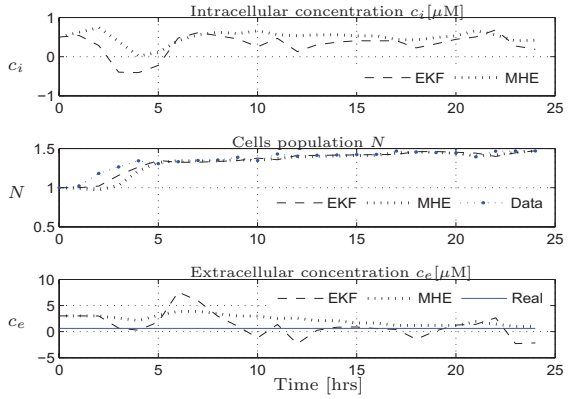


Fig. 5. *Concentration estimation of chromium (VI) corresponding to real $c_e = 0.62\mu M$. Bottom plot : Estimation of c_e converges to the real value (0.62μM) using both estimators. Top plot shows the estimation of c_i which is not measured. The middle plot shows the estimation of cell population that is measured.*

4.5 Concentrations estimation of mercury (II) chloride

Similar to the procedure adopted for estimating the concentration of chromium VI presented in the previous

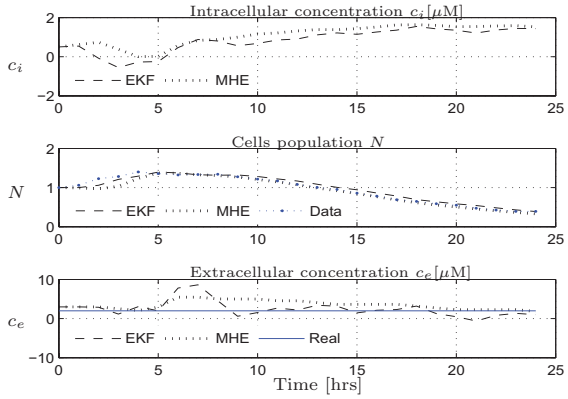


Fig. 6. *Concentration estimation of chromium (VI) corresponding to real $c_e = 1.97\mu M$. Bottom plot : Estimation of c_e converges to the real value (1.97μM) using both estimators. Top plot shows the estimation of c_i which is not measured. The middle plot shows the estimation of cell population that is measured.*

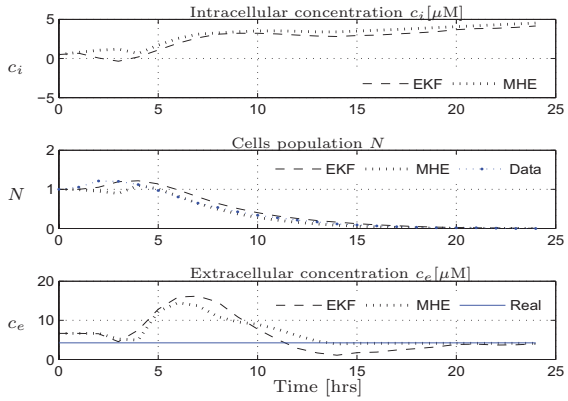


Fig. 7. *Concentration estimation of chromium (VI) corresponding to real $c_e = 4.25\mu M$. Bottom plot : Estimation of c_e converges to the real value (4.25μM) using both estimators. Top plot shows the estimation of c_i which is not measured. The middle plot shows the estimation of cell population that is measured.*

section, we aim here to estimate the concentration of mercury (II) chloride from the available data. The concentration c_e is added to the toxicity equation (4) of mercury (II) chloride as an augmented state similar to the augmented model (eqn. 11) for chromium (VI). We estimate the toxicant concentrations from three toxicity responses corresponding to the toxicant concentration c_e =(10.43; 22.35; 48.3)μM. The results presented in Fig. (8-10) show that both estimators (EKF and MHE) provide a good estimation of c_e while preventing negative concentration estimation when using MHE. This shows clearly the benefits of using constraints by MHE.

Our experience shows that tuning EKF is simpler. Using MHE requires a more careful tuning of a several parameters, namely, Q, R, Q_{-N_h}, horizon length (N_h), the constraints w_{min}, w_{max} and also the initial conditions. In addition, the tuning may vary from different experiments. The evolution of the estimation converges by using horizon length $N_h = 1$ for mercury (II) chloride case while at least $N_h = 2$ is needed for chromium IV case.

In the selection of the covariance, the Q matrix reflects the uncertainty of the state equations while the R matrix reflects the uncertainty in the measurement of CI due to other phenomena that also affect CI in addition to cell numbers.

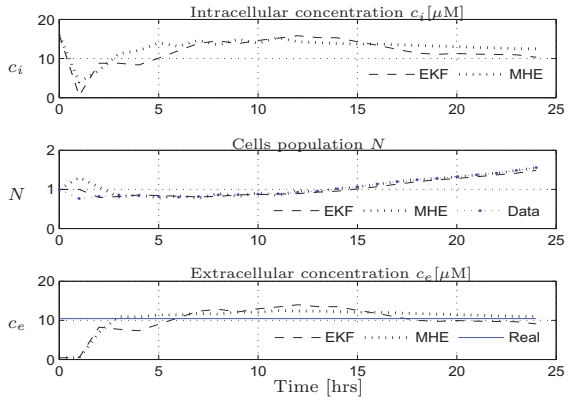


Fig. 8. *Concentration estimation of mercury (II) chloride corresponding to real $c_e = 10.43\mu M$. Bottom plot : Estimation of c_e converges to the real value (10.43μM) using both estimators. Top plot shows the estimation of c_i which is not measured. The middle plot shows the estimation of N that is measured.*

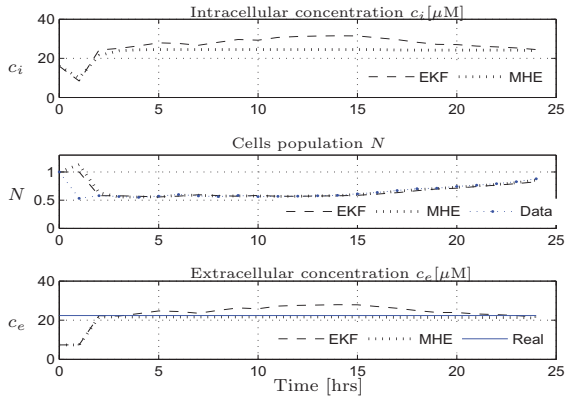


Fig. 9. *Concentration estimation of mercury (II) chloride corresponding to real $c_e = 22.35\mu M$. Bottom plot : Estimation of c_e converges to the real value (22.35μM) using both estimators. Top plot shows the estimation of c_i which is not measured. The middle plot shows the estimation of N that is measured.*

5. CONCLUSION

An early warning system for water supply is our main goal of the work presented. This includes an early determination of the presence of specific toxicants in water by on-line estimation of their concentrations. We use mainly MHE as an on-line estimation tool in this paper. Determination of the concentration is only one of the features of the aimed early warning system. This system will also include prediction of future evolution of toxicity response using only initial measurements and prediction of cells response when the concentration of a toxicant varies. Integrating all these prediction features is the ultimate goal. Intuitively, this includes also the development of toxicity mathematical models for other kinds of common water toxicants such as sodium arsenite [As (III)] for instance.

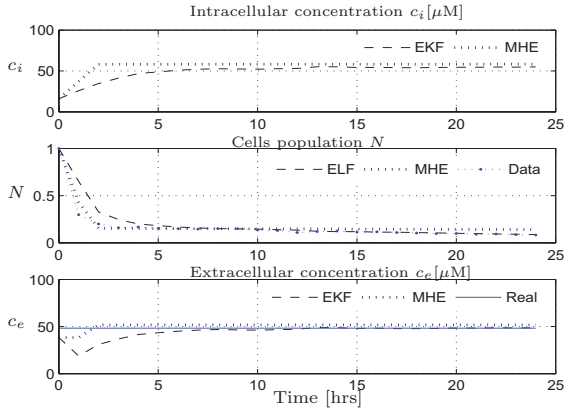


Fig. 10. *Concentration estimation of mercury (II) chloride corresponding to real $c_e = 48.3\mu M$. Bottom plot: Estimation of c_e converges to the real value (48.3μM) using both estimators. Top plot shows the estimation of c_i which is not measured. The middle plot shows the estimation of N that is measured.*

In addition, as has been discussed, the cell index also reflects other sophisticated physiological and pathological responses in addition to cell numbers. A model that considers other properties of the cell index will further improve on-line state estimation.

ACKNOWLEDGEMENTS

The authors gratefully acknowledge the support of the Natural Sciences and Engineering Research Council of Canada (NSERC).

REFERENCES

El-Kareh, A. and Secomb, T. (2005). Two-mechanism peak concentration model for cellular pharmacodynamics of doxorubicin. *Neoplasia*, 77, 705–713.

Eliaz, R., Nir, S., Marty, C., and Szoka, J. (2004). Determination and modeling of kinetics of cancer cell killing by doxorubicin and doxorubicin encapsulated in targeted liposomes. *Birches. J.*, 64, 711–718.

Haseltine, E. and Rawlings, J. (2005). A critical evaluation of extended kalman filtering and moving horizon estimation. *Ind. Eng. Chem. Res.*, 44, 2451–2460.

Huang, B. and Xing, J. (2006). Dynamic modeling and prediction of cytotoxicity on microelectronic cell sensor array. *Canadian Journal of Chemical Engineering*, 86, 393–405.

Rao, C. and Rawlings, J. (2002). Constrained process monitoring: Moving - horizon approach. *IChE Journal*, 48, 97–109.

Rao, C., Rawlings, J., and Lee, J. (October 2001). Constrained linear state estimation - a moving horizon approach. *Automatica*, 37, 1619–1628.

Rao, C., Rawlings, J., and Mayne, D. (2003). Constrained state estimation for nonlinear discrete-time systems: stability and moving horizon approximations. *Automatic Control, IEEE Transactions on*, 48, 246–258.

Walter, E. and Pronzato, L. (1996). On the identifiability and distinguishability of nonlinear parametric models. *Mathematics and Computers in Simulation*, 42, 125–134.

Wang, Q., Kim, D., Dionysiou, D., Sorial, G., and Timberlakeb, D. (2004). Sources and remediation for mercury contamination in aquatic systems - a literature review. *Environmental Pollution*, 131, 323–336.

Xing, J., Zhu, L., Jackson, J.A., Gabos, S., Sun, X.J., b. Wang, X., and Xu, X. (2005). Dynamic monitoring of cytotoxicity on microelectronic sensors. *Chem. Res. Toxicol*, 18 (2), 154–161.