Trajectory tracking in batch processes
using latent variable models

Jian Wan∗ Ognjen Marjanovic∗∗ Barry Lennox***

* Control Systems Center, School of Electrical and Electronic Engineering, The University of Manchester, Manchester, UK (e-mail: jian.wan@manchester.ac.uk)
** Control Systems Center, School of Electrical and Electronic Engineering, The University of Manchester, Manchester, UK (e-mail: ognjen.marjanovic@manchester.ac.uk)
*** Control Systems Center, School of Electrical and Electronic Engineering, The University of Manchester, Manchester, UK (e-mail: barry.lennox@manchester.ac.uk)

Abstract: The set-point tracking of certain process variable trajectories is often needed for the lower level control in batch processes so as to achieve desirable final product quality for the higher level control. In order to realize trajectory tracking successfully, process models should be known in advance. In fact, process models play an essential role in trajectory tracking. Due to the difficulty for developing first-principle models for batch processes, empirical models such as multi-way principal component analysis (PCA) and multi-way partial least squares (PLS) are increasingly used in practice. Trajectory tracking using multi-way PCA models has been proposed in the literature, where the underlying optimizations are performed in the latent variable space. This paper explores the corresponding application of multi-way PLS models for the task of trajectory tracking and compares it with the existing multi-way PCA model-based methods through benchmark case studies.

Keywords: Trajectory tracking; Principal component analysis; Partial least squares; Latent variable

1. INTRODUCTION

Due to the flexibility to cope with frequent changes in the demands and easy scale-up from laboratory procedures for manufacturing new products, batch/semibatch processing plays a significant role in the production of low-volume, high value-added products such as polymers, pharmaceuticals and specialty chemicals (Russell et al. (1998)). Batch processes are characterized by several features that pose significant challenges to control engineers. These characteristics include time-varying and nonlinear dynamics, large batch-to-batch variations, lack of reliable in-situ sensors and limited knowledge of analytical models that describe the dynamics of the processes. It is fair to say that the development of first-principle models for batch processes is usually challenging if not impossible in most occasions. Therefore traditional control approaches based on analytical models can hardly be implemented straightforwardly for batch processes. However, a multitude of process variables such as temperatures, pressures and flow rates are often routinely measured in modern manufacturing plants. Therefore, multivariate statistical process control (MSPC) methods, which are based on process history data to develop empirical models, become favorite or even standard approaches for process monitoring and control. Among them, multi-way principal compo-

* The authors are grateful for the funding provided by EPSRC with the grant number EP/G022445/1.
used for model building and the optimizations underlying MPC can be implemented in real-time space as well (Wan et al. (2010)). The approach was further extended to employ moving window PCA models so as to confront time-varying dynamics during batch processing (Golshan et al. (2010)). The role of the model structure for PCA model-based trajectory tracking was explored in Golshan et al. (2009), where the performance of batchwise and variable-wise unfolding for PCA models was compared. Compared to the success of PCA model-based trajectory tracking and PLS model-based batch endpoint control, there are few applications of PLS model-based control for trajectory tracking (Cimander and Mandenius (2004)).

This paper aims to extend PLS model-based control methods used for batch endpoint control to track a desirable trajectory and compare them with the existing PCA model-based method for trajectory tracking. The remainder of the paper is structured as follows. PCA and PLS are briefly introduced in Section 2. Section 3 describes the schemes of PCA and PLS model-based methods for trajectory tracking. Case studies for trajectory tracking using PCA and PLS model-based methods are given in Section 4. Finally, some conclusions and future remarks from this work are provided in Section 5.

2. LATENT VARIABLE MODELS

Latent variable models such as PCA and PLS are built from two-dimensional data while data collected from batch runs can usually be treated as a three-dimensional matrix of size $I \times J \times K$, where $I$ is the number of batches for which data are available, $J$ is the number of variables that are measured and $K$ is the number of samples collected during a batch run. Thus the first step to build latent variable models is to unfold the original data into a two-dimensional matrix.

2.1 Unfold batch data

Although there are various possibilities to unfold the data, the batchwise unfolding approach shown in Figure 1 is the most logical way for modeling difference among batches (Golshan et al. (2010)). In this approach, all the variables at different sampling times are unfolded as shown and each row in the unfolded matrix corresponds to one batch run history. A few latent variable scores summarize the major differences among the batches and the latent variable loadings capture all the time-varying dynamics of the batch process after performing PCA or PLS on the unfolded matrix.

2.2 Multi-way PCA

For multi-way PCA, all process variables are collected in the same three-dimensional matrix and performing PCA on the batchwise unfolded data of this three-dimensional matrix results in a reduced dimension latent variable model of the form:

$$X = TP^T + E,$$  (1)

where $X$ is the batchwise unfolded matrix of $I \times JK$, $T$ is a $I \times A$ matrix ($A \ll JK$) of latent variable scores that summarize the major differences among all batches, $P$ is a $JK \times A$ matrix of loading that captures the time-varying dynamics of the batch process, $E$ is the residual matrix since only $A$ latent variables are selected to account for most data variability through cross-validation (Diana and Tommasi (2002)). The data are mean-centered and scaled to be unit variance before performing PCA on it and thus the resulting latent variable model can be treated as a nonlinear model in the form of a locally linearized model of the covariance structure of the variables at every point in time (Golshan et al. (2010)).

2.3 Multi-way PLS

For multi-way PLS, process variables are divided into two groups. One stands for cause values such as measured process variable trajectories and the other stands for effect values such as measured batch quality variables (Marjanovic et al. (2006)). PLS works by selecting factors of cause variables in a sequence which successfully maximizes the explained covariance between the cause and effect variables. Performing PLS on the batchwise unfolded data of these two sets also results in a reduced dimension latent variable model of the form:

$$X = TP^T + E,$$  (2)

$$Y = UQ^T + F,$$  (3)

where $X$ is the batchwise unfolded matrix of $I \times JK$ for cause variables, $Y$ is the batchwise unfolded matrix of $I \times L$ for effect variables, $P$ of $JK \times A$ and $Q$ of $L \times A$ are the loading matrices for $X$ and $Y$, respectively. The scores $T$ and $U$ are related by a diagonal matrix $B$ of proper dimensions with $U = TB$. $T = BW$, where $W$ is the weight matrix. Finally, $E$ and $F$ are residual matrices.

Unlike the models built in (Lee and Lee (2007): Xiong
3. TRAJECTORY TRACKING USING LATENT VARIABLE MODELS

Based on the obtained latent variable models for batch processes, the future trajectories of manipulated variables can be deduced at each control decision point. They are to be implemented onto the process up to the next control decision point and the whole procedure is repeated in a receding horizon way. There are different strategies for trajectory tracking based on the type of latent variable models used in the receding horizon control.

3.1 PCA model-based trajectory tracking

The basic idea for PCA model-based trajectory tracking is to use the targeted trajectory to estimate the missing future manipulated variable trajectories (Flores-Cerrillo and MacGregor (2005); Golshan et al. (2009, 2010)). For the built multi-way PCA model based on the measured process variables $x$, the controlled process variables $y$ needed to be tracked and the manipulated variables $u$, the data for a new batch run can be rearranged as two parts at current sampling time $t_i$:

$$x_i^T = [x_{0→θ_i}^T, u_{0→θ_i→t_f}, y_{0→θ_i→t_f}, (y_{t_f+1→t_f})_{sp}]^T.$$  

(4)

where $t_f$ is the batch end time and $(y_{t_f+1→t_f})_{sp}$ is the targeted future process trajectory to be tracked. The loading matrix $P$ can also be decomposed into two corresponding parts for $x_1$ and $x_2$:

$$P^T = [P_1^T, P_2^T].$$  

(6)

Then the desired score for the new batch can be obtained through the following optimization problem:

$$\min_{t} (t^T - x_i^T P_1 - x_i^T P_2) Q_0 (t^T - x_i^T P_1 - x_i^T P_2)^T,$$  

(7)

s.t. $x_i^T = t^T P_2^T$

$$t_{min} \leq t \leq t_{max}$$

where $Q_0$ is the weight matrix. Other cost terms or hard constraints on the manipulated variables can also be added to the optimization formulation. Once the desired score is obtained from the optimization, the future manipulated variable trajectories can be deduced from $x_i^T = t^T P_2^T$ and they are to be implemented on the batch process in a receding horizon way. It is worthwhile to note that the cause-effect relationship between the manipulated variables and the controlled process variables has also been built implicitly in the obtained PCA model.

3.2 PLS model-based trajectory tracking

The control of batch product quality using PLS was proposed in Flores-Cerrillo and MacGregor (2004) and it can be extended for trajectory tracking by replacing the targeted batch product quality $y$ with the corresponding targeted process variable trajectories. Concretely, the cause variables for building the PLS model correspond to the measured process variables including the manipulated process variables and the effect variables $Y$ correspond to the controlled process variables to be tracked. Using the nominal future manipulated variable trajectories $(u_{0→t_f})_{nom}$, the score for a new batch run at time instant $t_i$ can be calculated as follows:

$$\hat{t}_i^T = [x_{0→θ_i}^T, u_{0→θ_i→t_f}, (u_{0→t_f})_{nom} (x_{θ_i+1→t_f})_{est}] W.$$  

(8)

where $(x_{θ_i+1→t_f})_{est}$ is estimated by missing data algorithms (Nelson et al. (1996)). The score is further adjusted according to the distance between the targeted and the predicted future trajectory $[(\hat{y} - y_{sp})]^T$:

$$\min_{\Delta t} (\hat{y} - y_{sp})^T \Gamma_1 (\hat{y} - y_{sp})^T + \Delta t^T Q_2 \Delta t + \gamma \tau^2,$$  

(9)

s.t. $\hat{y}^T = (\tau + t_i^p)^T BQ_2^T$

$$\tau^2 = \sum_{a=1}^{A} \frac{(\Delta t + t_{p,a})^2}{s_{p,a}^2},$$

$$\Delta t_{min} \leq \Delta t \leq \Delta t_{max}$$

where $Q_1$ and $Q_2$ are diagonal weight matrices, $\tau^2$ is the Hotelling's statistic, $s_{p,a}^2$ is the variance of the score $t_a$ and $\gamma$ is a weighting factor determining how tightly the solution is to be constrained to the region of the score space defined by past operations (Flores-Cerrillo and MacGregor (2004)). Based on the adjusted score value $t = t_i + \Delta t$, the future manipulated variables $u_{0→t_f}$ can be reconstructed according to the built PLS model. Denote $x_i^T = [x_{0→θ_i}^T, u_{0→θ_i→t_f}]$ and $x_{fut}^T = [u_{0→t_f}^T, x_{θ_i+1→t_f}^T]$, $P_1^T$ and $P_2^T$ are then their corresponding loading matrices while $W_1$ and $W_2$ are their corresponding weight matrices. The future manipulated variable trajectories to be implemented by receding horizon control can be reconstructed as follows (Flores-Cerrillo and MacGregor (2004)):

$$x_i^T = (t^T - x_i^T W_1) (P_2^T W_2)^{-1} P_2^T.$$  

(10)

Similarly, the evolving window PLS model-based control for batch product quality proposed in Wan et al. (2010) can also be extended for trajectory tracking, where a PLS model is built at each control decision point. The data of cause variables for building the PLS model are composed of the measured process trajectories up to the current time $x_{0→θ_i}$ and the entire manipulated variable trajectories $u_{0→t_f}$. The data of effect variables for building the PLS model is the controlled process variable trajectories to be tracked. Based on the built PLS model, the score for a new batch run at time instant $t_i$ can be calculated as follows:

$$\hat{t}_i^T = [x_{0→θ_i}^T, u_{0→θ_i→t_f}, (u_{0→t_f})_{nom}] W.$$  

(11)

It can be seen that the estimation of $(x_{θ_i+1→t_f})_{est}$ for a new batch run can be avoided by using evolving window PLS models instead. The following score adjustment as well as the reconstruction of future manipulated variable trajectories are similar to (9) and (10), respectively. At the next control decision point, the whole procedure is to be repeated by using a new PLS model built on the data of the corresponding cause and effect variables from the training batches.
4. CASE STUDIES

In order to assess and compare the performance of the addressed control approaches for trajectory tracking, a benchmark simulation is used. The simulator, called Pen-sim, is based upon a series of detailed mechanistic models that describe a penicillin fed-batch fermentation process (Birol et al. (2002)). The objective for each of the considered approaches is to track a desirable biomass concentration trajectory through manipulating substrate feed rates.

4.1 Data collection and model building

To enable the PCA/PLS models to fully capture the dynamics of the process, especially the cause-effect relationship between the substrate feed rate and the biomass concentration, the following process variables were collected hourly from simulated runs of 40 batches with a duration time of 200 hours for the fed-batch operation: aeration rate, agitator power, substrate feed temperature, substrate concentration, dissolved oxygen concentration, culture volume, pH, fermenter temperature and generated heat, substrate feed rate and biomass concentration. The substrate feed rate is the manipulated process variable while the biomass concentration is the process variable to be tracked. Filtered pseudo-random binary signals (PRBS) were appended to the nominal substrate feed rate of 0.045 l/h as well as aeration rate and agitator power for each batch so as to excite process dynamics and to generate batch-to-batch variations as well. The collected three dimensional data of \(40 \times 11 \times 200\) were then used for building the corresponding PCA/PLS models for trajectory tracking: for building the PCA model, all eleven process variables are unfolded together; for building PLS models, the biomass concentration is the effect variable and the rest ten process variables including the substrate feed rate are the cause variables. The built PCA model consists of 9 latent variables that account for over 70% data variability. The built PLS models consist of nine latent variables as well that fit over 70% of the \(X\) space and over 95% of the \(Y\) space.

4.2 Trajectory tracking using PCA/PLS models

The approaches for trajectory tracking are differentiated by the latent variable models used in the receding horizon control, i.e., the existing PCA model-based trajectory tracking termed FM-PCA, the extended PLS model-based trajectory tracking termed FM-PLS, and the evolving window PLS model-based trajectory tracking termed EW-PLS. Three scenarios were considered to evaluate their performance for trajectory tracking: track a nominal biomass concentration trajectory, track a modified biomass concentration trajectory and track a modified biomass concentration trajectory under additional disturbance on solution concentration of feeding substrate and measurement noises on biomass concentration and dissolved oxygen concentration. The ability of tracking modified trajectories and rejecting disturbance is important for a controller in practice since changing demands and varying disturbances often happen among/within batch runs.

The control results for tracking a nominal biomass concentration trajectory using FM-PCA, FM-PLS and EW-PLS are shown in Fig. 2, where the control decision points are at \(\theta_i = 20h, 70h\) and \(120h\). The nominal biomass concentration trajectory was generated by using the filtered PRBS around the nominal substrate feed rate of 0.045 l/h. The selected control decision points are just for demonstrating the process of renewing control inputs during the batch run and theoretically each sampling time could be a potential control decision point. The corresponding trajectories for the manipulated substrate feed rate using FM-PCA, FM-PLS and EW-PLS are shown in Fig. 3, where the open-loop substrate feed rate is kept constant at 0.045 l/h. Changing the filtered PRBS for the substrate feed rate to generate a different nominal biomass concentration trajectory, the simulation can be repeated for tracking the new nominal biomass concentration trajectory. The mean absolute trajectory tracking error \(\text{MAE}_Y = \frac{1}{N} \sum_{i=1}^{N} | y_i - y_{i,sp}| / N\), the mean absolute error for the changes in the manipulated process variable \(\text{MAE}_{\Delta u} = \frac{1}{N} \sum_{i=1}^{N} | u_{i+1} - u_i| / N\) as well as the standard deviation \(\sigma_u\) for one hundred testing batches are further listed in Table 1. It can be seen that all three control approaches can track the targeted biomass concentration trajectory. However, FM-PCA deduced a smoother substrate feed rate trajectory and thus it tracked the nominal biomass concentration trajectories better than FM-PLS and EW-PLS. This implies that FM-PCA captures the pattern of nominal operations through building the PCA model from training batches and thus it can replicate the nominal manipulated variable trajectory for any batch with nominal operating conditions. The replication is realized by using missing data algorithms to estimate future manipulated variable trajectories.

Modifying the targeted biomass concentration trajectory to be different from nominal operating conditions, the control results for these three methods are shown in Fig.
Fig. 3. Comparison of substrate feed rate trajectories in case of a nominal target

Fig. 4. Comparison of tracked biomass concentration trajectories in case of a modified target

4, where the control decision points are also at $\theta_i = 20h, 70h$ and $120h$. The corresponding trajectories for the manipulated substrate feed rates are shown in Fig. 5. Similarly, the mean absolute trajectory tracking error $\text{MAE}_y = \sum_{i=1}^{N} |y_i - y_{i,sp}|/N$, the mean absolute error for the changes in the manipulated process variable $\text{MAE}_\Delta u = \sum_{i=1}^{N} |u_{i+1} - u_i|/N$, as well as the standard deviation $\sigma_u$ for one hundred testing batches with modified biomass concentration trajectories are further listed in Table 2. It is obvious that the ability to adapt to the modified biomass concentration trajectory has changed for these three control approaches, where FM-PLS and EW-PLS perform better than FM-PCA because of the optimizations underlying them to minimize the difference between the predicted trajectory and the targeted trajectory while FM-PCA only uses the targeted trajectory data to estimate the future manipulated process variables.

Due to varying property of raw materials, the solution concentration of feeding substrate during the batch run can be different from its nominal value of 600$g/l$. Considering a step change of substrate solution concentration from 600$g/l$ to 580$g/l$ at 10th hour and measurement noises on biomass concentration as well as dissolved oxygen concentration, the control results for these three methods are shown in Fig. 6, where the control decision points are at $\theta_i = 20h, 70h$ and $120h$. The corresponding trajectories for the manipulated substrate feed rates are shown in Fig. 7. The mean absolute trajectory tracking error $\text{MAE}_y = \sum_{i=1}^{N} |y_i - y_{i,sp}|/N$, the mean absolute error for the changes in the manipulated process variable $\text{MAE}_\Delta u = \sum_{i=1}^{N} |u_{i+1} - u_i|/N$, as well as the standard deviation $\sigma_u$ for one hundred testing batches of the same kind are listed in Table 3. It can be seen that EW-PLS performs slightly better than FM-PCA and FM-PLS in terms of tracking errors. This can be attributed to the looser fixation of nominal patterns in evolving window PLS models as the future process variables are not estimated using missing data algorithms as in FM-PCA and FM-PLS models. Nevertheless, the tracking errors are all relatively large due to the plant-model mismatch in case of disturbances.

5. CONCLUSIONS

This paper has investigated the use of multi-way PLS models for trajectory tracking and also their comparison with the existing PCA model-based method for trajectory tracking.
Two types of PLS models are studied: FM-PLS model and EW-PLS model. The existing FM-PCA has excellent performance for tracking nominal trajectories as it captures the pattern of nominal operating conditions. However, its ability to track modified trajectories is worse than FM-PLS and EW-PLS for lack of minimizing the difference between the predicted trajectory and the targeted trajectory. The EW-PLS method shows some advantage to track disturbed trajectories because of its looser fixation of nominal patterns in evolving window PLS models. However, the obtained manipulated variable trajectories for EW-PLS tends to be more oscillatory than FM-PCA and FM-PLS. Potential measures to eliminate the oscillation and more benchmark examples are to be explored in the future work to fully expand the potential of tracking trajectory using late variable models.

ACKNOWLEDGEMENTS

The authors would like to acknowledge The Process Modeling, Monitoring, and Control Research Group at Illinois Institute of Technology who generously provided the source code for their Pensim simulator.

REFERENCES


