REAL-TIME OPTIMIZATION OF DISTILLATION COLUMN VIA SLIDING MODES

Andrei Yu. Torgashov, Kyung-Chul Park, Ho-Chul Choi, Yun-Kee Choe

Technical Team, Samsung Fine Chemicals Co., Ltd.,
190, Yeochun-dong, Nam-ku, Ulsan, 680-090, Korea
E-mail: andrei.torgashov@samsung.com

Abstract: The real-time optimization (RTO) system of distillation column has been proposed using sliding modes. On the basis of formulated optimization problem, the parameters for the distillate composition controller are so selected that the condition for the occurrence of sliding modes holds in order to provide a search procedure. An investigation of the transient performance of RTO-system under feed composition disturbances indicated a stable tracking for the shifted optimal distillation operating points.

Keywords: Distillation columns, Sliding-mode control, Optimization, Uncertain dynamic systems, Disturbance rejection.

1. INTRODUCTION

The maintaining an optimal steady-state of the distillation column in chemical industry presents the important task for control system. Due to the large-scale distillation dynamic model, the optimal controller can not be found using, for example, principle of maximum technique or another analytical method. The high dimensionality of distillation model is inciting the researchers to develop the more efficient RTO-systems in which the real model complexity can be considered as uncertainty.

Nowadays the widespread approaches for process optimization in real-time are divided on the two main groups: model-based (Duyfjes and Grinten, 1973; Forbes and Marlin, 1996; Cheng and Zafiriou, 2000) and direct search strategies (Rastrigin, 1974). The separate place among the methods of the optimal process operation search has a self-optimizing control guaranteeing a determination only a sub-optimal solution with the minimum criterion losses (Skogestad, 2000).

The main drawback of the model-based RTO-systems is the model uncertainty. In this case the identification procedure is required for the updating model parameters. Moreover, it is difficult to predict the final time of identification under continuously acting disturbances. In this issue the optimization process can lose a convergence property and may be unstable. Some results concerning the model-based RTO-systems design are generalized in the work (Zanin, *et al.*, 2000). It was shown that the integration a model-based strategy with Model

Predictive Control (MPC) sometimes does not provide a successful search, for example, using Sequential Quadratic Programming (SQP).

The direct search strategy does not depend from the accuracy of the model. The sequence of steps are organized in the direction of the criterion gradient descent. At the final steps, the stable auto-oscillations are observed near the criterion extremum neighbourhood. As noted in the work (Rastrigin, 1974), such RTO-systems fall into two main types. The first one is the backspacing -based systems. During the backspacing-based search, the switching of the optimizable variable (x) is fulfilled if the criterion derivative $dQ(x)/dx$ (where $Q(x)$ - criterion) reaches the small given value. The second type of the direct search systems is the RTO strategy based on the synchronous detection. Using the synchronous detection technique, the harmonic excitation signal (for instance, a sinusoid wave) arrives at the plant input and the criterion value is obtained. The phase between input and output is detected and the magnitude of $dQ(x)/dx$ is estimated by the corresponding manner in order to get an appropriate moving into extremum.

The common lack of the direct search approaches is the necessity to measure a gradient of function. In the real industrial conditions it is often possible to get an inaccurate gradient of function. This is the cause of the continuos or unstable search procedure. The work [7] shows that the sliding modes can be applied for the static optimization problem solution and it is unnecessary to measure criterion derivatives.

The present paper proposes the RTO-system based direct search via sliding modes. The profit is considered as criterion for developed RTO-system. The feed composition disturbances generate a drift of the optimal distillation steady-states. It will be demonstrated that the RTO-system is insensitive for the disturbances influence in the sliding mode and capable to track the optimal process performance.

2. RTO PROBLE FORMULATION

In this section we consider a steady-state optimization problem of the distillation column adopted from the work (Skogestad, 2000) because of its simplicity and convenience for a demonstration of developed RTO-system. The two product binary distillation column is examined (fig.1). The relative volatility of the separated compounds has the constant value α=1.12. The concentration of the light component comprises 99.5% in the distillate. The bottoms purity specification is not given. The distillate (D) and vapor boil-up (V) are taken as manipulated variables and the overhead product purity is adjusted by the distillate flow (fig.1). The column has one degree of freedom (i.e. V). Table 1 contains the nominal steady-state parameters of the distillation column. The profit function is formulated in the following way

$$P=p_D D+p_B B-p_F F-p_V V,$$

where the prices are given in [\$/kmol]: p_D=20; p_B=10-20x_B; p_F=10; p_V=0.1. Consider the case when there are no restrictions on the D and V. The RTO problem can be stated as the tracking task for the minimum of the function J=-$P(V)$ under feed composition disturbances z_F (fig. 2). Notice that such disturbance type has the significant and nonlinear affecting on the shifting of the criterion extremum as compared with a feed flow rate or liquid phase in the feed disturbances.

Table 1 Nominal operating point of the distillation column

Parameter	Value
Feed flow rate	F = 1 kmol/min
Vapor boil-up	V = 15.6381 kmol/min
Distillate flow rate	D = 0.6381 kmol/min
Feed composition (light component)	z_F = 65 %
Distillate composition	x_D = 99.5 %
Bottoms composition	x_B = 4.1575 %
Liquid phase in the feed	q_F = 1
Total number of trays	N = 112
Feed tray number	f = 39

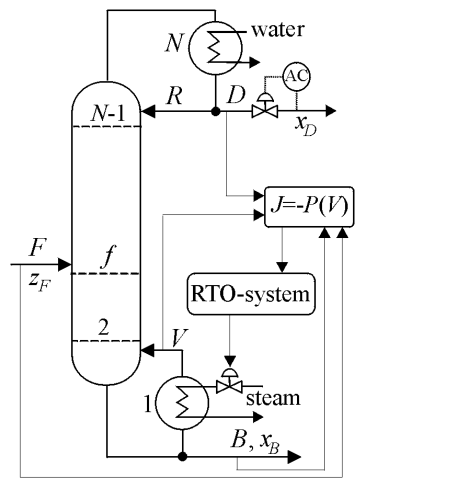

Fig. 1. The sketch of distillation column and RTO-system.

3. SLIDING MODES BASED RTO-SYSTEM FOR DISTILLATION COLUMN

As discussed in the work (Korovin and Utkin, 1974), the sliding modes can be successfully applied in the solving of static optimization problems. The framework of this approach is as follows. The plant output is compared with a certain specially selected reference input which is a monotonically decreasing function of time. Input actions of the plant are obtained from the difference between the output and the reference input and should reduce this difference to zero. As a result the plant output follows the monotonically decreasing setpoint and reaches a minimum. The main feature of this kind of tracking system is that the value and the sign of the varying local gain are unknown. The RTO-system must provide a trend in the plant output variation such that its output should always decreases by following up the reference input.

The proposed scheme of the sliding modes based RTO-system is depicted on the fig. 3 and described by the following equations (taking into account our statement of optimization problem)

$$J=-P(V),\ V=u,\ u=u_0\mathrm{sign}(\sigma_1\sigma_2),$$
$$\sigma_1=\varepsilon,\ \sigma_2=\varepsilon+\delta,\ \varepsilon=g-J, \tag{1}$$

where $\dfrac{dg}{dt}=-\rho+h(\sigma_1\sigma_2)$;

$$h=\begin{cases}-M,\ if\ \sigma_1-\Delta>0\ and\ \sigma_2>0\\ 0,\ if\ (\sigma_1+\Delta)(\sigma_2-\Delta)<0\\ +M,\ if\ \sigma_1<0\ and\ \sigma_2+\Delta<0\end{cases}.$$

The switching elements are shown in the fig.4.

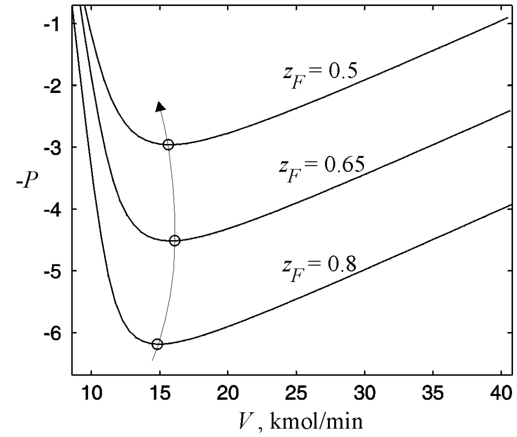

Fig. 2. Shifted optimum of the criterion under various z_F and constant value of x_D=99.5 %.

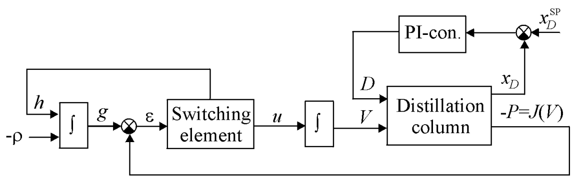

Fig. 3. RTO-system based on the sliding modes for distillation column with the distillate composition control loop (PI-controller).

The parameters of the RTO-system (fig.3) are chosen in accordance with the instructions cited in the article (Korovin and Utkin, 1972). The drift of the steady-state operating point is provided under disturbances rejection by the PI-controller in order to meet the following inequality

$$\left|\frac{\partial J(V)}{\partial V}\right|u_0\le\left|\frac{\partial J(V)}{\partial t}\right| \tag{2}$$

Here, the parameters of the PI-controller are defined so that response time of the x_D-D control loop is significantly faster as compared with the P-V optimization loop. The transfer function of PI-controller has the form (subject to eq. (2))

$$W_{\mathrm{PI}}(s)=5\frac{0.1s+1}{0.1s}. \tag{3}$$

For ensuring the sliding modes in the RTO-system, the following values of the variables in (1) were derived:

$$\rho=0.0006,\ u_0=0.0025,\ M=5,\ \delta=1,\ \Delta=0.1.$$

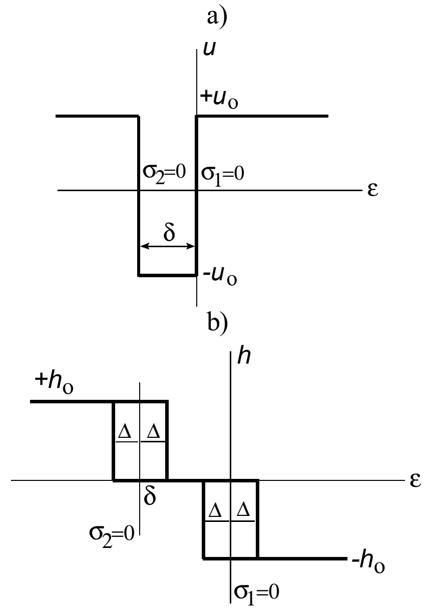

Fig. 4. Switching elements of RTO-system.

The dynamic distillation model of the considered column consists from 110 differential equations. The liquid and vapor molar flows inside the column are assumed to be constant. The liquid flows hydrodynamics is neglected. This model is so simple but contains the main features of the process and gives the possibility to analyze the proposed RTO-system. The disturbances variations schedule is presented in the Table 2 for three time intervals.

Figure 5 depicts the simulation results of the transient performance for RTO-system according to the feed composition disturbances in Table 2. It should be pointed that the system operating in the sliding mode is low-sensitivity for the acting feed composition disturbances and provides the automatic tracking of the optimal process steady-state.

Table 2 Operating points of distillation column under various feed compositions (RTO results)

No	Time (min)	z_F (kmol/kmol)	$-P^{opt}$ [$/min]	V^{opt} (kmol/min)
1	0-4000	0.65	-4.52	16.05
2	4000-8000	0.8	-6.19	15.80
3	8000-12000	0.5	-2.96	14.70

4. CONCLUSION

The application of sliding modes in RTO-system design for a distillation column has been proposed in the present paper. There are two main advantages of the developed RTO-system:

1) the missing of the criterion derivatives measurements for the establishing of the search procedure;
2) the independence from the uncertainty of a first-principle model as in the model-based RTO-systems.

The optimum operating points tracking task becomes more complex under disturbances influence because it is difficult to measure the vector-gradient function at the non-stationary conditions. Therefore, the RTO-system parameters (1)-(2) selection represents a try-and-error technique.

It was shown that the proposed RTO-system ensured a stable convergence toward the optimal distillation column steady-states even though the large initial deviations of the optimized variable from the V^{opt} and various feed composition disturbances were involved.

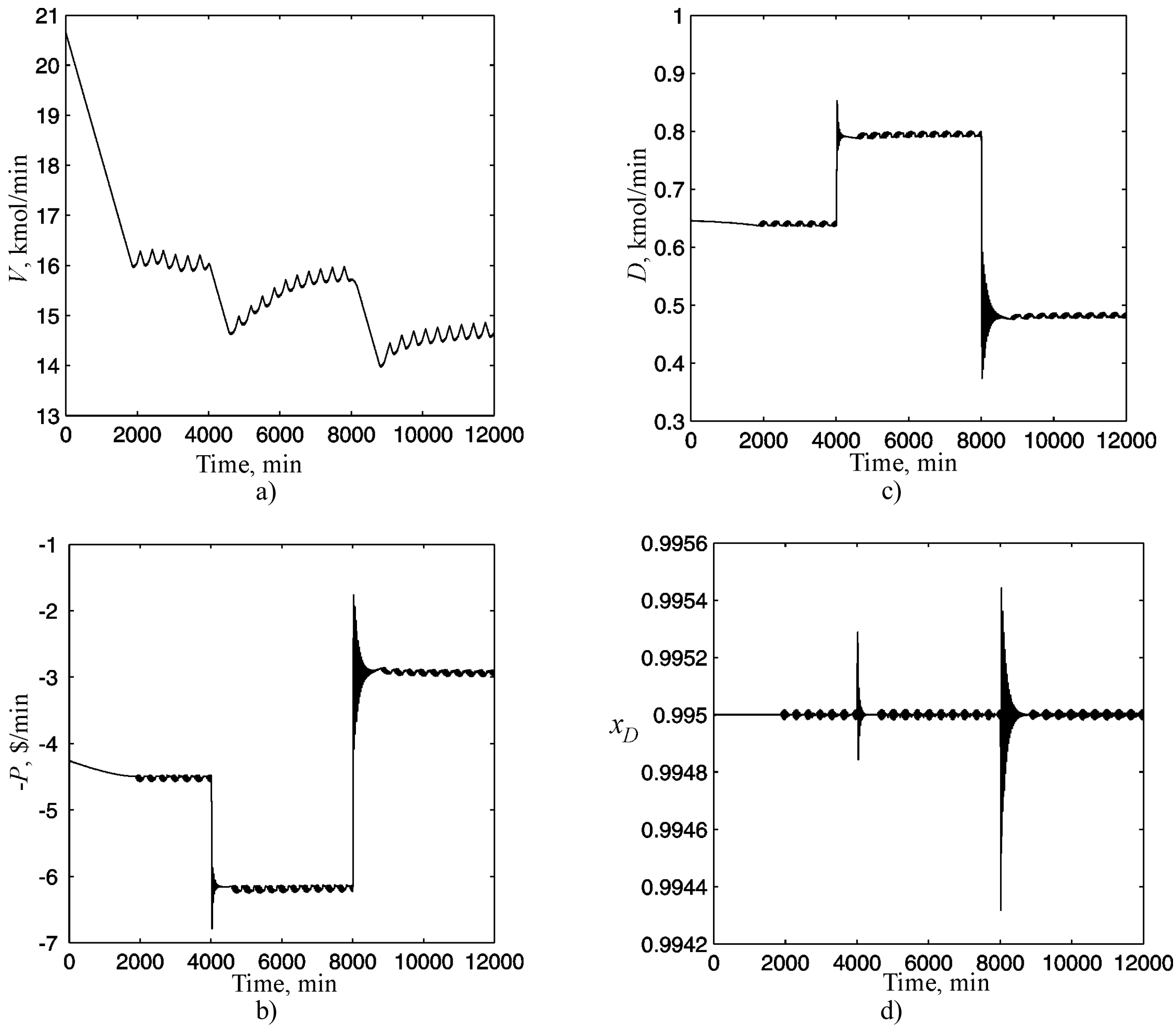

Fig.5(a-d). Optimal operating points tracking for distillation column by the sliding modes based RTO-system.

REFERENCES

Cheng, J-H. and E. Zafiriou (2000). Robust model-based iterative feedback optimization for chemical plants, *Proc. IFAC-symposium Adchem'2000, Pisa, Italy*, 887-892.

Duyfjes, G. and P. M. E. M. van der Grinten (1973). Application of a mathematical model for the control and optimization of a distillation plant. *Automatica*, Vol. 9, 537-547.

Forbes, F. and T. Marlin (1996). Design cost: a systematic approach to technology selection for model-based real-time optimization systems, *Comp. Chem. Engng*., Vol. 20, 717-734.

Korovin, S.K. and V.I. Utkin (1972). The use of the sliding modes in static optimization problems. *Aut. Remote Control*, 50-60.

Korovin, S.K. and V.I. Utkin (1974). Using sliding modes in static optimization and nonlinear programming. *Automatica*, Vol. 10, 525-532.

Rastrigin L.A. (1974). *Optimizing control systems*. Nauka, Moscow. (in Russian)

Skogestad S. (2000). Plantwide control: the search for the self-optimizing control structure. *Journal of Process Control*, Vol. 10, 487-507.

Zanin, A.C., Tvrzska de Gouvea, M. and D. Odloak (2000). Comparing different RTO strategies for the FCC catalytic converter, *Proc. IFAC-symposium Adchem'2000, Pisa, Italy*, 803-808.

A RECEDING OPTIMIZATION CONTROL POLICY FOR PRODUCTION SYSTEMS WITH QUADRATIC INVENTORY COSTS

SONG Chunyue[1] **WANG Hui**[2] **LI Ping**[1]

(1: *Institute of Industrial Process Control*, 2: *Institute of Systems Engineering Hangzhou 310027*)
(*E-mail: cysong@iipc.zju.edu.cn*)

Abstract: For stochastic disturbance, such as stochastic demands and breakdown of the system, the production systems is presented as a piecewise deterministic process model. At any given time only one type of product can be produced by the system. A setup (with setup time and cost) is required if production is to be switched from one type of product to another. Preventive maintenance activity is performed for reducing the aging of the system, and the jump rates of the system state depend on the aging of the system. The objective of the problem is to minimize the costs of setup, production, maintenance and the quadratic costs of inventory. The decision variables are a sequence of setups and the production and preventive maintenance plan. According to two time horizons, the original problem is decomposed into two sub-problems. The asymptotic optimality solution of the original problem is constructed via receding optimize the sub-problems. Simulation results show the feasibility of the proposed approach in practice.

Keywords: Optimal production, Equipment aging, Dynamic programming, Receding optimization

1. INTRODUCTION*

Many works focused on the optimal production of manufacturing systems, and the outstanding efforts have been made in the past years (see, e.g., Yan and Zhang, 1997; Sethi and Zhang, 1994, 1995; Gershwin, 1989 and the references therein). However, there are very few lectures casting light on the production systems in processing industry such as a miner water production line. In the system, it is needed to wash pipeline when the production process of a type of mine water product is finished and switched to another type of product. Furthermore, to avoid deteriorating, the duration time of the products being stocked will not be long and their value reduces with time. On the other hand, the failure-prone equipment will be aging during its using. Recently there have been some scholars focusing on the field. Shu and Perkins (2001), and Boukas and Liu (2001) discussed manufacturing systems with deteriorating items. But in their lectures, setup and aging were not discussed. Boukas (1987) and G. Liberopoulos and M. Caramanis (1994) discussed the aging of the machine that affects the frequency of its failure. Boukas and Haurie (1990) and Boukas et al. (1994) discussed preventive maintenance of flexible manufacturing systems considering the machine age function.

In most stochastic production systems with unreliable machines, the optimal production planning is an extremely difficult problem, both theoretically and computationally. The nature of the production system provides it with hybrid dynamic systems characteristics, i.e., possessing both continuous variable dynamical systems characteristics and discrete event dynamical systems characteristics. Based on above, the optimal production of these systems is far more difficult than that of manufacturing systems. And some conclusions about the optimal production of manufacturing systems could not be directly applied to that of the production systems. In the paper, the production system is presented as a piecewise deterministic process model with controlled Markov disturbance. The aging of system and preventive maintenance is taken into consideration and the optimal production of the production systems is discussed. Here, according to two time horizons, the original problem is decomposed into two sub-problems. And the asymptotic optimality solution of the original problem is constructed via receding optimized the sub-problems.

* This work is supported by National Natural Science Foundation under Grant 69804009 and National High Technique Research Development Project under Grant 2001AA414240.

The paper is organized as the follows. Section 2 presents production system description, dynamic model and objective function. Section 3 simplifies the original problem into two sub-problems, and a receding algorithm framework is also given. Section 4 presents an example exposing basic advantages of the method, and the last Section concludes.

2. DESCRIPTION OF THE PROBLEM

The production system consisting of a set of unreliable equipments can produce n different types of product P_i, i=1, … , n with only one at any given time. Moreover, a setup (with setup duration and setup cost) is required if production is to be switched from one type of product to another. The equipment is subject to random failure and repairs. For reducing the aging of the equipment, the maintenance activity involving lubrication, routine adjustments, etc, will be preformed when the equipment is being used. It is assumed for i, j=1, … , n and $i \neq j$, constants $\theta_{ij} \geq 0$ and $K_{ij} \geq 0$, which denote the setup duration and cost of switching from production of P_i to P_j, respectively. Moreover, for any i, j, k=1, … , n, $i \neq j$ and $j \neq k$, $max\{\theta_{ij}, K_{ij}\}>0$, $\theta_{ij}+\theta_{jk}-\theta_{ik} \geq 0$ and $K_{ij}+K_{jk}e^{-\rho\theta ij}-K_{ik}>0$. If $i=j$, then $\theta_{ij}=K_{ij}=0$. Here $0<\rho<1$ denotes the discount rate.

2.1 *THE DYNAMIC MODEL OF THE SYSTEM*

For $t \geq 0$, let $x_i(t) \in R^1=(-\infty,\infty)$, $u_i(t) \in R^+=[0,\infty]$, and $z_i(t) \in R^+$ denote the surplus, production rate, and the rate of demand for product P_i at time t, i=1, … , n. X, U, and Z are used to denote vectors $[x_1(t), x_2(t), …, x_n(t)]^T \in R^n$, $[u_1(t), u_2(t), …, u_n(t)]^T \in R^{+n}$, and $[z_1(t), z_1(t), …, z_n(t)]^T \in R^{+n}$, respectively, where A^T denotes the transpose of a vector (or a matrix) A. $h(t)$ is used to represent the age of the equipment at time t, $h(t) \in R^+$. The inventory/shortage levels and equipment age of the system are described by the following dynamic differential equations:

$$\begin{cases} \dot{X}(t) = F(\alpha, U(t), Z(t)) = U^{\alpha}(t) - Z(t) \\ \dot{h}(t) = f(u^{\alpha}(t), v^{\alpha}(t)) \qquad (X(0), h(0)) = (X_0, h_0) \end{cases} \tag{1}$$

Where $F(\cdot,\cdot)=[F_1, F_2, …, F_n]^T$, $U^{\alpha}=[u^{\alpha}_1, …, u^{\alpha}_n]^T$ and $u^{\alpha}(t) = \sum_{i=1}^{n} u_i^{\alpha}(t)$. $v_i^{\alpha}(t) \in R^+$, $u_i^{\alpha}(t)$ is the maintenance rate of the system, the instantaneous production rate of type of product P_i, respectively, at time t with the equipment state $\zeta(t)=\alpha$(defined later). The function f in Eq.(1) represents the effect of the production rate $u^{\alpha}(t)$ on the equipment age and $f(u^{\alpha}(t))=0$ when the equipment is under repair (Boukas and Haurie, 1990). The unreliable equipment states can be classified as (i) breakdown, denoted by state 0; (ii) maintenance, denoted by state 1; (iii) operational or setup, denoted by state 2. Under operational or maintenance state, any type of product can be produced; under breakdown state, nothing is produced. Let $\zeta(t)$ denote the state process of the equipment, and let $E=\{0,1,2\}$ be the state space of the process $\zeta(t)$, $\zeta(t) \in E$.

Let $q_{\alpha\beta}(h(t))$ be the jump rate of the process $\zeta(t)$ from state α to state β at time t. These jump rates are defined by

$$P[\zeta(t+dt)=\beta|\zeta(t)=\alpha] = q_{\alpha\beta}(h(t))dt+o(dt) \tag{2}$$
$$P[\zeta(t+dt)=\alpha|\zeta(t)=\alpha] = 1+q_{\alpha\alpha}(h(t))dt+o(dt) \tag{3}$$

Where $\lim_{dt \to 0} o(dt)/dt=0$, $q_{\alpha\alpha}(\cdot)=-\sum_{\beta \neq \alpha} q_{\alpha\beta}(\cdot)$. It is assumed that the jump rate $q_{\alpha\beta}(h(t))$ are bounded and satisfy the following conditions: $|q_{\alpha\beta}(h(t))-q_{\alpha\beta}(h'(t))| \leq C|h(t)-h'(t)|$, $\forall\, h(t)$, $h'(t) \in R$, for some constant C and $|q_{\alpha\alpha}| \geq c_0>0$, $q_{\alpha\beta}(h(t)) \geq 0$.

When the equipment has a breakdown, it goes through a repair process. The repair time is usually random and described by the repair rates. The equipment repaired is considered renewed, i.e., the age of the equipment is reset to 0. Since f=0 when the equipment is under repair, for convenience, the age $h(t)$ is reset to 0 at the beginning instead of the end of the repair process. Since inventory control is considered, the outcome will not be influenced, as during the repair process the equipment age remains a constant and its value does not influence the inventory level. Thus, according to our notation, if there is a jump from state α to state β, then the age function $h(t)$ jumps to $h'(t)=\beta h(t)$. According to the above, the following holds

$$h(t+\Delta t) = \begin{cases} 0, & \text{If } \zeta(t+\Delta t)=1 \text{ and } \zeta(t-\Delta t) \neq 1; \\ a(t), & \text{Other.} \end{cases} \tag{4}$$

where $\Delta t>0$ is small enough.

2.2 *THE COST FUNCTION AND CONSTRAINTS*

Over the infinite horizon, we are concerned with the optimality problem of finding a production control policy that minimizes the following cost function:

$$J(i, X, s, \Xi, U(\cdot), h(\cdot), v(\cdot), \alpha) = \int_0^s e^{-\rho t} G(X(t), 0, 2) dt$$
$$+ E\left(\int_s^{\infty} e^{-\rho t} G(X(t), U(t), \zeta(t)) dt + \sum_{l=0}^{\infty} e^{-\rho \tau_l} K_{i_l i_{l+1}} \right) \tag{5}$$

Where s denotes the remaining setup time, $0 \leq s \leq \theta_{ij}$. The decision variables are the rates of production $U(\cdot)$ over time and a sequence of setups denoted by $\Xi=\{(\tau_0, i_0 i_1), (\tau_1, i_1 i_2), …\}$, where a setup (τ, ij) is defined by the starting time τ and a pair ij denoting that the equipment was already set up to produce P_i and is being switched to be able to produce P_j. Let

G(X(t),U(t),ζ(t)) denote the instantaneous cost function of the surplus, repair and maintenance. We denote

$$G(X(t), U(t),\zeta(t))=\sum_{i=1}^{n} c^{+}_{i}(x_i^{+})^2+c^{-}_{i}x_i^{-}$$
$$+c_r ind\{\zeta(t)=0\}+c_m ind\{\zeta(t)=1\} \qquad (6)$$

Positive surplus is supposed to incur a holding cost of c_i^{+} per unit commodity per unit time, while the negative a cost of c_i^{-}, with $c_i^{+}>0$, $c_i^{-}>0$. $x_i^{+}:=max(x_i,0)$, $x_i^{-}:=max(-x_i,0)$. Where c_r and c_m denote cost parameter of repair and maintenance respectively, and $c_r >> c_m$. They are nonnegative constants. $ind\{\zeta(t)=\alpha\}$ is the indicator function of set $\{\zeta(t)=\alpha\}$. The quadratic instantaneous cost function is a useful cost approximation for systems where products are perishable or may become obsolete, as well as systems with storage-space competition.

For $t\geq 0$, the production constraints are given as follows:

$$\begin{cases} 0 \le u_i(t) \le \zeta(t) r_i, \quad i=1,2,\cdots,n \\ u_j(t)=0, \quad j \ne i \\ 0 \le v(t) \le v_{\max,} \end{cases} \qquad (7)$$

Where r_i denotes the maximum production rate of P_i and v_{max} is a constant. Let $U(\alpha)$, a close subset of R^{+n}, denote the production rate control constraints, $\forall \alpha\in E$. Any measurable function $U(t)$ defined on $U(\alpha)$, for each $\alpha\in E$, is called an admissible control. The set $\Theta=\{U(t):t\geq 0\}$ is an admissible policy. The admissible control function $U(t)$ is supposed to be piecewise continuous in t and continuously differentiable with bounded partial derivatives in X. $U(t)$ is a feedback admissible control which can react to the current state. Feedback controls are of practical importance because they will adjust any unfavorable deviation of the state from the targeted position at any time and hence render a better performance, especially when uncertainties or disturbances are presented in the system.

Let ($X(t)$,α,ij) denote the system state at time t, and the space of the system state is $R^n\times E\times\{ij|i, j=1, 2, \ldots, n, i\neq j\}$. The problem is to find an admissible decision $(\Xi,U(\cdot))\in\Omega=(\Xi,\Theta)$ that minimizes $J(i, X, s, \Xi, U(\cdot), h(\cdot), v(\cdot), \zeta(t))$ which is subject to Eq.(1), (7).

3. SIMPLIFIED MODEL FOR THE PRODUCTION SYSTEM

For the large-sized production systems and the presence of some stochastic events, it may be quite difficult to obtain exact optimal feedback policies to run these systems, both theoretically and computationally. One way to cope with these complexities is to develop methods of hierarchical control of these systems. Gershwin, Sethi and Zhang reached many significant conclusions in the direction. Here, the original problem is decomposed into two sub-problems according to the occur frequency of events. Based on the nature of production systems the setup is treated as a typical controllable event. In detail, the setup series and production rate are gotten in the static problem without considering unreliability of the system, and real-time production rate and maintenance rate are solved in the dynamic problem according to the aging of the system. Both static and dynamic problems are discussed on receding horizon. And the asymptotic control policy is composed of the solutions of static and dynamic problems.

3.1 *THE STATIC PROBLEM*

Without losing generality, let s=0, and P_i denote the initial product being produced. Over the finite horizon [0,T] the objective function can be written as the following without considering the dynamic properties of the system

$$J(i,X,0,\Xi,U(\cdot),2)=\int_0^T e^{-\rho t}G(X(t),U(t),2)dt+\sum_{l=0}^{m} e^{-\rho\tau_l}K_{i_l i_{l+1}}$$
$$=\int_0^{T_1} e^{-\rho t}G(X(t),U(t),2)dt+\int_{T_1}^{T_1+\theta_{12}} e^{-\rho t}G(X(t),0,2)dt+e^{-\rho T_1}K_1$$
$$+\int_{T_1+\theta_{12}}^{T_1+\theta_{12}+T_2} e^{-\rho t}G(X(t),U(t),2)dt+\cdots$$
$$+\int_{\sum_{i=0}^{m-1}\theta_{ii+1}+\sum_{i=0}^{m-1}T_i}^{\sum_{i=0}^{m-1}\theta_{ii+1}+\sum_{i=0}^{m}T_i} e^{-\rho t}G(X(t),U(t),2)dt$$
$$+\int_{\sum_{i=0}^{m-1}\theta_{ii+1}+\sum_{i=0}^{m}T_i}^{\sum_{i=0}^{m}\theta_{ii+1}+\sum_{i=0}^{m}T_i} e^{-\rho t}G(X(t),0,2)dt+e^{-\rho(\sum_{i=0}^{m-1}\theta_{ii+1}+\sum_{i=0}^{m}T_i)}K_i \qquad (8)$$

Where $T_0=K_0=\theta_{01}=0$, and T denotes the terminate time when the whole production ends, and T_i denotes the terminate time when the ith type of production is over, for i=1, 2, …, m. It is obvious that $T\to\infty$ as $m\to\infty$.

In any optimal policy, there is always some nonzero time for producing the intended product after the completion of each setup (Sethi and Zhang, 1995), i.e., $T_i>\Delta>0$. Since T_i responds to the inventory $X(T_i)$, T_i as a new state variable is incurred. Let $\text{И}_i=[T_1, T_2, \ldots, T_i]$, i=1, 2, …, k, $T_i\in R^{+}$, then И_i denotes the production time series before i+1th setup. Let the optimal decision of Eq.(8) be $V_{k-i}[j, X(i), \text{И}_i]$ when the initial state is $(j, X(i), \text{И}_i)$, then a Bellman equation of Eq.(8) can be gotten by dynamic programming

$$V_{k-i}[j,X(i),\text{И}_i]=\min_{u_j(i),\Xi}\{J(j,X(i),\text{И}_i)+V_{k-(i+1)}[l,X(i+1),\text{И}_{i+1}]\} \qquad (9)$$

The solution of Eq.(9) is the optimal production of the system over the finite horizon $[0,T]$ when unreliability is not considered. And the setup series Ξ^*, $x^*_i(T^*_1)$ and T^*_1 will be conveyed to the dynamic problem as an expected value.

3.2 *THE DYNAMIC PROBLEM*

Since the optimal setup times, production rate and the optimal inventory have been gotten by the static planning level, furthermore, the optimal production duration of the initial product being produced is also determined, this paragraph is focused on how to get the real-time production rate and maintenance rate when considering the unreliability and the aging of the system. For using receding algorithm, only given type of product is discussed in the following. Most of papers on optimal control of (non)flexible manufacturing systems consider the demand a constant, which is in fact assumed constant over a short term but not accurate over a long term. It is a piecewise function at least. Demand rate can be gotten by analyzing orders for commodity or the historical data with predictive method. Demand rate is treated as a variable here. According to dynamic property and aging of the equipment, the real time control is carried out using receding horizon control policy.

Without losing generality, let s=0, and the discrete model of the dynamic system on the horizon can be written as follows

$$X_i(k+1)=X_i(k)+U_i(k)-Z_i(k) \tag{10}$$

$$h(k+1)=h(k)+f(u(k), v(k)), \quad u(k)=\sum_{i=1}^{n} u_i(k) \tag{11}$$

$$J(i,X,0,\Xi^*,U(\cdot),h(\cdot),v(\cdot),\alpha)= E\left(\sum_{k=0}^{T/t_p} e^{-\rho k} G(X(k),U(k),\zeta(k)) + \sum_{l=0}^{m} e^{-\rho\tau_l} K_{i_l i_{l+1}}\right) \tag{12}$$

Where t_p denote receding horizon. To get the optimal policy of Eq.(12), a Hamilton-Jacobi- Bellman equation must be solved, and Sethi and Zhang reached many significant conclusions in the direction via viscosity solution. The asymptotic optimality solution of the original problem can be gotten by receding optimized the two sub-problems. And the receding algorithm framework is described as follows:

Step1: initialization: assume product P_i is to be produced. $z_i(0)=z_0$, $x_i(0)=x_0$;
Step2: use the given θ_{ij}, K_{ij} to compute: Ξ^*, $x^*_i(T^*_1)$ and T^*_1 by s.(1), (7), (9);
Step3: solve Eq.(10), (11), (12) according to Ξ^* to get u^*_i and v^*_i;
Setp4: if T^*_1 is reached, go to step 2, else go to step 3;

4. SIMULATION OF EXAMPLES

The performance of the policy is shown with examples including following specifications: n=2, and ρ =0.9, z_1=z_2=0.4, $\alpha \in E$={0,1,2}. The other parameter is shown in Tab.1. It is assumed that f is linear. For various initial conditions $X(0)$, the optimal production control is listed in Tab.2. Only optimal production durations of the initial product being produced are listed. Simulation shows that the optimal production control policy is of region switching structure, and of hedging point policy when f is linear.

The trend of the objective function $J(\cdot,\cdot)$ changing with T_1 is illustrated in Fig.1 as T_i is optimal, i=2, 3, …, k. And T_1 is the optimal production duration when $J(\cdot,\cdot)$ is its minimum. Five curves in Fig.1 agree with those five examples in Tab. 2. The simulation results show that different initial conditions respond to different optimal production durations of the initial product being produced. In examples 1, 2, 4 (the sold line), since product P_2 is not sufficient, sometimes even deficient, the policy shortens the optimal production duration of P_1, which are different from Ex. 3(the dotted line). In Ex. 3, P_2 is sufficient, which prolongs the optimal production duration of P_1, but each product is sufficient in Ex. 5(the dashed line), which makes the system produce nothing. The results also agree with hedging point policy.

Table.1 Parameters of the system

θ_{12}	θ_{21}	K_{12}	K_{21}	C_r	C_m
0.65	0.75	1.25	1.15	1.86	0.23

Table.2 Results of simulation

Ex.	$x_1(0)$	$x_2(0)$	C_1^+	C_1^-	C_2^+	C_2^-	T_1	min $J(\cdot,\cdot)$
1	-2.5	-2.0	0.5	3.0	0.6	3.0	1.40	14.2745
2	-1.5	0.0	0.5	3.0	1.0	3.0	3.20	4.5565
3	-2.5	1.5	0.5	3.0	1.0	3.0	5.40	7.5554
4	0.0	0.0	1.0	3.0	1.0	3.0	0.90	1.7482
5	2.0	2.5	1.0	3.0	1.0	3.0	0.0	5.1151

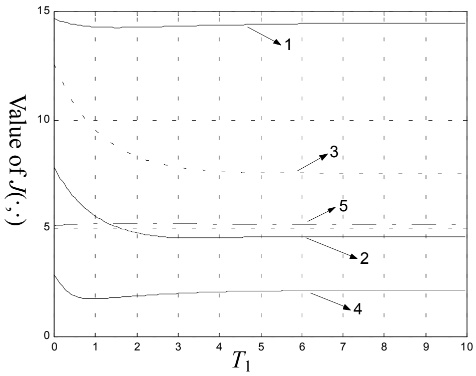

Fig. 1 Tendency of value $J(\cdot,\cdot)$ to T_1

Over a finite time, the policy not only keeps the system run at the least cost but perfectly satisfies the demand. Moreover the policy makes the production satisfy the customers in sum and balances all types of the products, keeping the inventory in low level.

5. CONCLUSIONS

In the paper, the age function, which affects the occur frequency of the failure, is incurred to the objective function with quadratic inventory costs. Based on the nature of the production systems, setup is treated as a typically controllable event in the static sub-problem without considering the details of the setup. Furthermore, the production duration T_i of one type of product as a new state variable is introduced to the problem. In the dynamic sub-problem, based on the age function of the equipment, the production rate and maintenance rate in real-time are gotten by receding algorithm.

The policy decreases the complexities of the original problem, i.e., reduces the stochastic optimal production control problem of multi-dimension vector to the determinist optimal production control problem of multi-dimension vector, and keeps near to the stochastic problem by sliding on one dimension, which renders the receding algorithm feasible, more accurate and real-time. Simulation results show the merits. The age function of the equipment and quadratic inventory costs make the objective near to the practice. However, the optimized solution is not optimal over all globe, but an asymptotic solution. And the receding control policy can decrease the drawback.

REFERENCES

Boukas E. K. (1987), Commande optimale stochastique appliquée aux systèmes de production, *PhD Thesis*, Ecole Polytechnique de Montreal, Université de Montreal.

Boukas E. K. and A. Haurie (1990), Manufacturing flow control and preventive maintenance: A stochastic control approach, *IEEE trans. Automat. Contr.*, Vol. 35, pp.1024-1031.

Boukas E. K., Q. Zhu and Q. Zhang (1994), Piecewise deterministic Markov process model for flexible manufacturing systems with preventive maintenance. *Journal of Optimization Theory and Application*, Vol. **81**(2), pp.259-275.

Boukas E. K. and Z. K. Liu (2001), Manufacturing systems with random breakdowns and deteriorating items, *Automatica*, Vol. 37, pp.401-408.

Filar J. A. and A. Haurie (2001), A two-factor stochastic production model with two time scales, *Automatica*, Vol. 37, pp.1505-1513.

Gershwin S. B. (1989), Hierarchical flow control: A framework for scheduling and planning discrete events in manufacturing systems. *Proceedings of the IEEE*, Vol. 77(1), pp.195-209.

Liberopoulos G. and M. Caramanis (1994), Production control of manufacturing systems with production rate-dependent failure rates. *IEEE Transactions on Automatic Control*, Vol. **39**(4), pp.889-895.

Sethi S. P. and Q. Zhang (1994), Hierarchical production planning in dynamic stochastic manufacturing systems: Asymptotic optimality and error bounds, *J. Mathemat. anal. appli.*, Vol.181, pp.285-319.

Sethi S. P. and Q. Zhang (1995), Hierarchical production and setup scheduling in stochastic manufacturing systems, *IEEE trans. Automat. Contr.*, Vol. 40, pp.924-930.

Shu C. and J. R. Perkins (2001), Optimal PHP production of multiple part-types on a failure-prone machine with quadratic buffer costs, *IEEE Trans. Automat. Contr.*, Vol. 46(4), pp.541-549.

Yan H. and Q. Zhang (1997), A numerical method in optimal production and setup scheduling of stochastic manufacturing systems, *IEEE trans. Automat. Contr.*, vol. 42, pp.1452-1455.

HARD REAL-TIME CORBA (HRTC) FOR PROCESS CONTROL SYSTEMS [2]

Santos Galán, Manuel Rodríguez, Ricardo Sanz [1]

ASLAB, Universidad Politécnica de Madrid, Spain

Abstract: Control systems for process plants are complex applications running in several interacting computers with varying degrees of integration. The construction, deployment and maintenance of the software system is a difficult problem and distributed object oriented technology offers a good way to deal with it. The open standard CORBA provides flexible middleware capable of integrating complex applications in heterogeneous environments, but was originally designed with large business applications in mind and is not perfectly suited for the construction of control systems. Even with recent advances in the real-time specification for CORBA, it is only suitable for soft real-time applications and do not deal with the tight requirements of closed control loops. In this paper, the building of a process control testbed to identify requirements for CORBA control systems, with both predictable and event-driven transports, is presented. The benefits of such technology are discussed.

Keywords: Real-Time, CORBA, Process Control, Distributed Object Computing

1. INTRODUCTION

Most present-day plant-wide control systems are very complex, constituted by diverse hardware and software components which interact with each other. With the incorporation of intelligent sensors, the computers reach even the lower level of the control hierarchy. They are also distributed systems, different tasks run on different processors (computers, networks interfaces, PLC's...) and common resources are shared between processors. Distributed systems are designed to improve performance and increase system reliability in order to meet timing, resources and concurrency constraints on each node.

Control systems have been traditionally separated into several levels:

(1) Field level. This level is dedicated to the instruments (sensors and actuators) and basic regulatory control. It is communicated via fieldbus.
(2) Process control level. This level takes over the advanced and supervisory control, including local optimization. It is communicated via an Ethernet based protocol.
(3) Business level. The upper level is dedicated to global optimization, scheduling and planning. It is communicated via Ethernet.

Although these levels have been always present in the process industry the control implementation has been evolving along the years. From the first direct digital control where all the devices were connected separately to the control room where the control was centralized to a single computer; to the traditional Distributed Control System (DCS) implementation where several devices are linked to a controller and there are several distributed controllers that are connected to the DCS

[1] corresponding author, ricardo.sanz@aslab.org
[2] The project is funded by the European Commission as IST-37652, "HRTC, Hard Real-Time CORBA".

console; to the future where the control is totally distributed to field with the loops in individual devices. Nowadays we are still in the traditional DCS but migrating slowly to the future

To implement these coming distributed systems efficiently and with enough flexibility, middleware seems to be the most appropriate tool to simplify the task. The construction, deployment and maintenance of the software system is an extremely difficult problem. Even though there are no silver bullets, object oriented technology offers a good way to build complex systems and when they are running in several, networked computers, distributed object technology has been demonstrated as a feasible way to cope with this complexity while keeping costs under control.

CORBA (Common Object Request Broker Architecture, (OMG, 2000; OMG, 1999; OMG, 1998)) is an open standard which provides developers of distributed systems with a flexible middleware capable of integrate complex applications in heterogeneous environments. It should not matter the programming language or operating system chosen to be part of the system, CORBA makes it possible through a feature called interoperability.

In the global Distributed Object Computing (DOC) landscape, CORBA is a well known framework for the construction of modularised, object oriented, distributed applications. It was designed from the perspective of surpassing heterogeneity barriers and provide support for modularity and reuse. CORBA, however, was originally designed with large business applications in mind and is not perfectly suited for the construction of embedded control applications. This has changed recently because the RT (Real-Time) SIG (Special Interest Group) inside the OMG is very active in the development of specifications for this field: Real-time CORBA has found its place into mainstream CORBA specifications. This makes CORBA a specification that deals with real-time issues from the very core (a real difference from other distributed objects technologies).

Some people in the process industry consider that CORBA is an alternative replacement of OPC, but this is based on a lack of understanding of both technologies. CORBA is not a service but a middleware technology that happens to be better than COM (the software under OPC). Particular services —like OPC— can be built and delivered atop of it. In fact there is an OMG specification that provides a complete replacement with enhancements of OPC servers (it is called HDAIS). CORBA is much wider in scope than OPC and technologically more sound and powerful than COM. For example, CORBA specifies mechanisms for real-time behavior or fault tolerance that is basic for the construction of control applications.

CORBA is used by the process industry. RiskMan (Sanz *et al.*, 2000) (based on the ICa (Sanz *et al.*, 1999*a*) broker) is a system for emergency management in a chemical complex with nine plants (see Figure 1). The system supports the whole life-cycle of emergencies: prevention, detection, firing, diagnosis, handling, follow-up and cancellation. The application is composed by a collection of CORBA objects running on heterogeneous platforms (VAX/VMS, Alpha/UNIX, x86/Windows NT) performing an heterogeneous collection of functions: expert systems, user interfaces, wrappers of real-time plant databases, data filters based on fuzzy rules, predictors based on neural networks, etc.

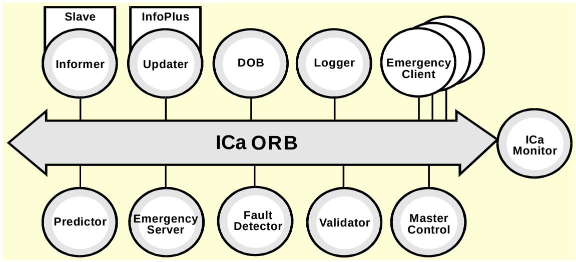

Fig. 1. Some of the CORBA objects that compose the RiskMan application. Informer and Updater are wrappers of external systems.

HRTC uses a Real-time Object Request Broker (developing a prototype implementation of a hard real-time network transport), to build a Process Control Testbed (PCT) that address issues of hard real-time composability in heterogeneous applications, identifying the requirements for hard real-time distributed control systems using CORBA technology. The testbed experiments should be able to prove some of the benefits that such distributed object computing can bring to the process control field.

The paper is organized as follows: Section 2 summarizes the basic concepts of Real-Time CORBA. Section 3 discusses the changes proposed in CORBA needed to achieve hard real-time performance. Section 4 presents the Process Control Testbed and the experiments to identify requirements for HRTC. Section 5 concludes the paper discussing the expected benefits of the technology.

2. CORBA, REAL-TIME AND REAL-TIME CORBA

2.1 What is CORBA?

CORBA is an open standard that allows programmers to specify interfaces as contracts between servers and clients (these roles are classic in distributed applications). These interfaces are specified using a language called IDL (Interface Definition Language). IDL is used naturally with object oriented programming languages, which map IDL types to their native types after passing IDL files through an IDL compiler. Basically, an interface is an object service contract, implemented by a server, and a way to decouple it form its implementation so that changes to an implementation do not involve a whole re-compilation of the system.

CORBA's key entity is called Object Request Broker (ORB). An ORB is a software bus capable of transmitting messages through a network, from clients to servers, in a transparent way. Clients invoke server methods through a proxy or stub; the ORB locates the server, transmits the invocation from client to server and after the server executes the operation, brings back the results to the client. The servant is a server object which implements an IDL interface and is plugged to the ORB via an object adapter; the most common being the Portable Object Adapter (POA).

2.2 What is Real-Time?

A good definition of the field of hard real-time systems is provided by Douglas Locke (Locke, 2000) from TimeSys:

"What is real-time? A real-time system (as defined by IEEE) is a system whose correctness includes its response time as well as its functional correctness. In other words, in a real-time system, it not only matters that the answers are correct, but it matters when the answers are produced. Note that by this definition, systems requiring a defined Quality of Service are usually real-time systems, although they might not use those words to describe themselves.

What is hard real-time? Hard real time means that the system (i.e., the entire system including OS, middleware, application, HW, communications, etc.) must be designed to GUARANTEE that response requirements are met. It doesn't matter how fast the requirements are (microsecond, millisecond, etc.) to be hard real-time, just that they MUST be met EVERY TIME."

Some applications like cellular phones, web servers or digital television need real-time behaviour but in most cases they do not need hard real-time. Other applications like aircraft or process control are presently built as soft real-time but in its very nature, they pose hard real-time requirements to systems developers.

2.3 Real Time CORBA

RT CORBA is an extension of the CORBA standard whose intention is aiding the design of real-time distributed applications. RT CORBA defines CORBA priorities which have corresponding native priorities on each operating system. The interface PriorityMapping is responsible of this conversion. There are two priority models of distributed priority handling:

- Client Propagated: The server honours the priority requested by the client, who sends it along with the invocation.
- Server Declared: The server establishes its own priorities and ignores client requested priorities.

What a client can do using RT CORBA is:

- Set a priority or band of priorities for a given connection.
- Obtain a private transport (non demultiplexed connection) to a servant, so that the connection is not shared with other clients.
- Set a timeout on an invocation.

What a server can do:

- Manage execution of threads through Threadpool interface. Threads can be preallocated (so that server is limiting the number of incoming requests with the possibility of buffering requests that cannot be dispatched) and partitioned in priority lanes responsible of managing requests with a priority bounded to a certain range.
- Select a priority model (Client propagated vs. Server declared).
- Create a Mutex so that the client can prevent other server threads to access certain piece of code of the server.

Both client and server can select a communication protocol and configure certain protocol parameters.

RT CORBA also defines a service called Scheduling Service. This service is designed to work in a closed environment, where clients and servers can be considered a static set, with fixed priorities. Scheduling service provides global scheduling policies, associating names with scheduling parameters. RT CORBA 1.0 does not provide dynamic scheduling. A new extension (RT CORBA 2.0) specifically addresses dynamic scheduling.

3. HARD REAL-TIME CORBA

The CORBA object model (and the development processes and tools associated with it) is extremely adequate for the construction of complex distributed applications and hence the interest in extending it to be useful in the real, embedded control domain. But there is a problem. Present day CORBA specifications are suitable only for soft real-time applications. CORBA and its extension RT CORBA are not fully suitable to implement these systems because:

- They have only been designed to build systems with soft real time requirements.
- CORBA lacks of a real-time interoperable protocol, necessary to integrate control and real time systems. Neither GIOP nor IIOP are reliable or predictable enough.
- The Scheduling Service is incomplete, can not be dynamically reconfigured and does not provide a wide range of scheduling algorithms.
- Most real time systems are also embedded ones. There is an effort called Minimum CORBA to build a small ORB, tailoring it to fit in embedded systems, but this seems to exclude RT CORBA which increases ORB size.
- Interface specification needs to be extended to express temporal issues.

The analysis of hard-real time requirements posed by CORBA-based distributed control systems shows the necessity to develop theory and technology for hard-real time applications, extending the set of CORBA specifications with interfaces that deal with hard real-time issues.

3.1 What does CORBA need to be Hard Real-Time CORBA?

Some authors claim that advances in real-time distributed object computing can be achieved only by systematically pinpointing performance bottlenecks; optimising the performance of networks, ORB endsystems, common services, and applications; and simultaneously integrating techniques and tools that simplify application development. We believe that a sound engineering approach to system design is also necessary.

Building hard real-time systems with stringent constraints requires the election of an appropriate environment which includes:

- Choosing real-time operating systems for critical nodes: with real-time I/O subsystems and with real-time scheduling.
- Choosing predictable (usually high-speed) network interfaces, communication protocols or industrial backplanes suitable for real-time applications like ATM, CAN, VME, switched fabric, fieldbuses, etc. They must be highly predictable and provide flexibility of control (because TCP/IP, GIOP or IIOP are not very suitable).

RT CORBA 1.0 is thought to be used with static systems, where processes, clients, servers and tasks are perfectly known and let us determine the best policies for our system. This not flexible enough and does not provide needed reconfiguration capabilities. RT CORBA 2.0 includes dynamic scheduling but it is still not enough.

In our opinion, CORBA needs to be extended in certain aspects because:

- CORBA requires a deterministic transport and a reliable and interoperable RT protocol, whose QoS parameters can be modified through CORBA interfaces.
- RT Scheduling need support for dynamic algorithms and support for advanced feedback scheduling.
- CORBA interfaces must be specified not only in the value domain but also in the temporal domain.
- Another problem is global time synchronization. Deployment over time triggered platforms can provide enhancements in distributed time.
- Meeting hard real time requirements includes validating them. This can be done with interceptors, but it is a time consuming way. Maybe some other method should be used to do this.

4. PROCESS CONTROL TESTBED

In order to identify (mainly hard real-time) requirements for distributed control systems and perform experiments in conditions of systems heterogeneity and legacy integration a Process Control Testbed is used. Experiments will be done using conventional IIOP and a new real-time protocol.

Figure 2 shows the complete topology of the proposed testbed. This final structure should be reached in several stages of increasing difficulty where different experiments are run.

The PCT tries to represent the basic characteristics of a process plant control system network with advanced features not found in current designs, like the flat two control networks (Ethernet and TTP/C (TTTech Computertechnik, 1999; Kopetz, 1997)) where all the elements are linked. Several instruments (sensors and actuators) are connected to a (actual or simulated) process plant in three different ways:

(1) Trough a typical industrial distributed control system (DCS), in this case the TPS from Honeywell that constitutes a legacy system in this context, with its own controller and user interface. The TPS communicates with the Ethernet control network.

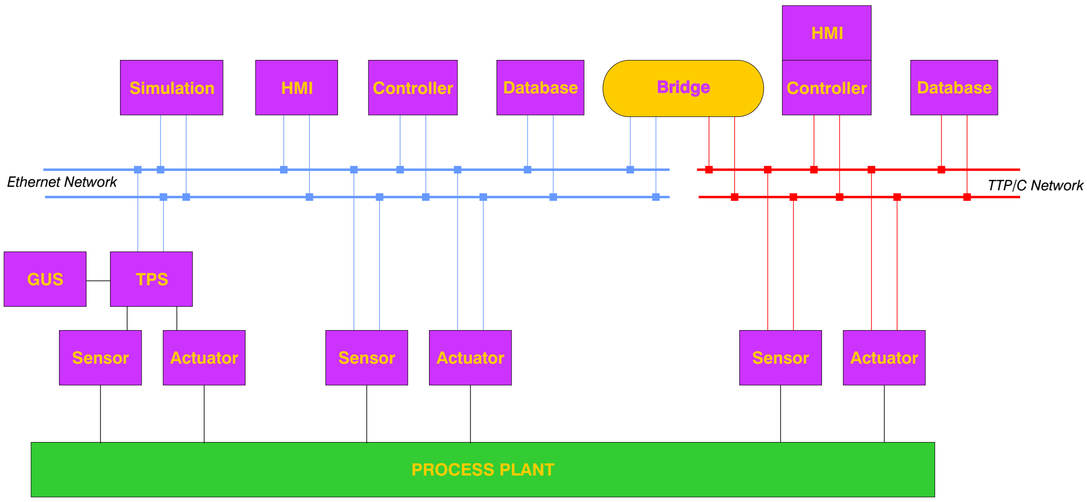

Fig. 2. Schema of the Process Control Testbed

(2) Directly connected to an Ethernet control network.
(3) Directly connected to a time-triggered network (TTP/C).

Apart from the TPS monitoring and controlling devices, both networks include controllers, human-machine interfaces (HMI) and history databases. This is not the typical configuration in industrial practice, where separate networks are used. Finally, one or several simulation nodes are included on the Ethernet network. The Ethernet and time-triggered networks communicate through a brigde.

4.1 Closing control loops through networks

A simple regulatory control loop with three components: Sensor, Actuator and Controller, built as independent nodes connected through the Ethernet and the TTP/C networks are tested (figure 3). Also there should be two additional nodes: HMI and History Database. For the TT network the HMI and the controller are in the same node.

In the experiment, an operating elemental process plant, such as a neutralization tank with a pH sensor is controlled by the adition of a reactant with a volumetric pump. The time series of values of the process variables are recorded on the history database and shown on real time on the HMI. The operator can change the setpoint through this HMI node.

This is apparently a simple experiment but its success will demonstrate the use of CORBA for control systems. When designing a distributed real-time system, scheduling of common resources is a key problem. For distributed control systems where control loops are closed over a communication network or a field bus, the network can be a bottleneck.

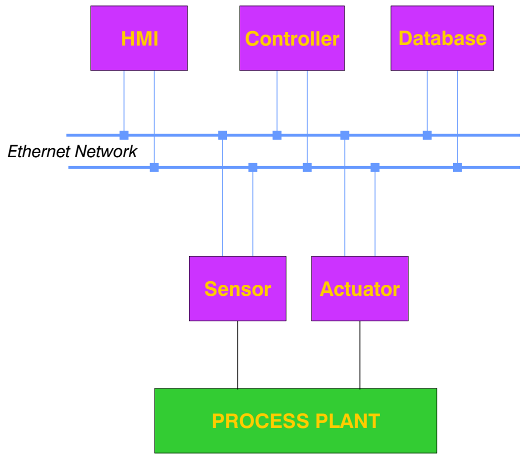

Fig. 3. Control loop with CORBA components over Ethernet

A similar experiment will be run over a TTP/C network, comparing both results.

4.2 Legacy system integration

This experiment is aimed to demonstrate the possibility of integrating legacy systems in a CCS (figure 4). As indicated above, a Honeywell TPS, with its graphical user interface (GUS) is used.

4.3 Interaction of simulation objects with control agents

A test for the integration of heterogeneous components over the same network is the connection of simulation modules that interact with other objects in the system (figure 5). The use of mathematics models in control and monitoring functions is continuously increasing and control systems should accommodate easily this components.

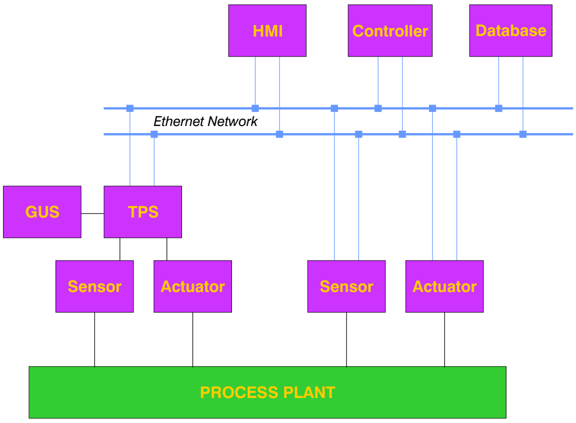

Fig. 4. Schema of the PCT for integration of DCS (legacy system)

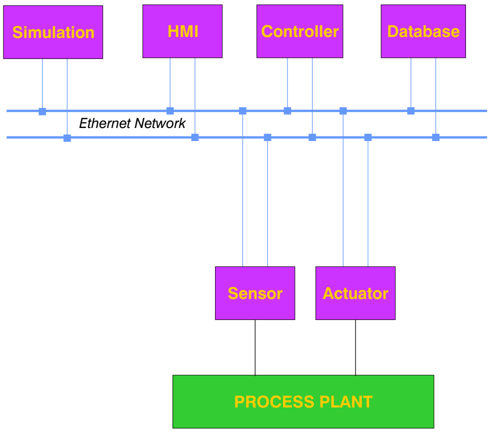

Fig. 5. Integration of simulation objects in control network

5. CONCLUSIONS

CORBA provides the middleware capable of integrating complex applications but it needs to be upgraded to hard real-time to be applied in process control systems.

Perhaps the main question is: Why do we need the integration provided by HRTC in process control systems? Beyond many obvious answers (simplicity of flat network, use of heterogeneous components like optimization or simulation, vendor independence, reduction in cost, etc.) we would like to stress one: The modular approach fostered by CORBA will let us develop true modular control systems.

The second point we want to mention is design freedom. Design freedom is necessary in the complex control systems domain to explore alternative controller designs. Excessively restrictive technologies will collapse - unnecessarily - dimensions of the controller design space (Shaw and Garlan, 1996). This is, for example, the case of some fieldbus technologies that support several slaves but only one master. While design restrictions (in the form of prerequisite design decisions) simplify development, they sacrifice flexibility. Can we get both, simple development and flexibility? The key are frameworks where design dimensions are still open even when pre-built designs are available. To continue the example of the fieldbus, the one-master/several-slaves approach is one type of pre-built, directly usable, design; but the underlying fieldbus mechanism should allow for alternative, multi-master designs. This can be done by means of the development of agent libraries that provide predefined partial designs in the form of design patterns (Sanz *et al.*, 1999*b*), and a transparent object-oriented real-time middleware like the one proposed in HRTC. This approach will let developers construct their own agencies to support their own designs.

REFERENCES

Kopetz, Hermann (1997). *Real-Time Systems – Design Principles for Distributed Embedded Applications*. Kluwer Academic Publishers.

Locke, Douglas (2000). Real-time linux – what is it, why do you want it, how do you do it?. At6090565653.html. LinuxDevices.com. http://www.linuxdevices.com/articles/.

OMG (1998). A discussion of the object management architecture. Technical report. Object Management Group. Falls Church, USA.

OMG (1999). Real-time CORBA adopted specification. Technical Report orbos/99-06-02. Object Management Group. Falls Church, USA.

OMG (2000). Common object request broker architecture and specification. Release 2.4. Technical report. Object Management Group. Falls Church, USA.

Sanz, Ricardo, Miguel J. Segarra, Angel de Antonio and José A. Clavijo (1999*a*). ICa: Middleware for intelligent process control. In: *IEEE International Symposium on Intelligent Control, ISIC'1999*. Cambridge, USA.

Sanz, Ricardo, Miguel J. Segarra, Angel de Antonio, Fernando Matía, Agustín Jiménez and Ramón Galán (1999*b*). Design patterns in intelligent control systems. In: *Proceedings of IFAC 14th World Congress*. Beijing, China.

Sanz, Ricardo, Miguel Segarra, Angel de Antonio, Idoia Alarcón, Fernando Matía and Agustín Jiménez (2000). Plant-wide risk management using distributed objects. In: *IFAC SAFEPROCESS'2000*. Budapest, Hungary.

Shaw, Mary and David Garlan (1996). *Software Architecture. An Emerging Discipline*. Prentice-Hall. Upper Saddle River, NJ.

TTTech Computertechnik (1999). *Specification of the TTP/C Protocol*.

A DISAGGREGATION TECHNIQUE FOR THE OPTIMAL PLANNING OF OFFSHORE PLATFORMS

Carvalho, M.C.A.; Pinto, J.M

Dept Chemical Engng, University of Sao Paulo, Sao Paulo 05508-900, Brazil;
Dept Chemical Engng and Chem., Polytechnic University, Brooklyn NY 11201.

Abstract: There is a great incentive for developing systematic approaches that effectively identify strategies for planning oilfield complexes. This paper proposes an MILP that relies on a reformulation of the model proposed by Tsarbopoulou (UCL M.S. Dissertation, London, 2000). Moreover, a disaggregation technique is applied to the MILP. A master problem determines the assignment of platforms to wells and a planning subproblem calculates the timing for fixed assignments. Results show that the decomposition approach generates optimal solutions for instances of up to 145 wells and 64 platforms in 10 discrete time periods that otherwise could not be solved with a full-scale model.

Keywords: mathematical models, integer programming, discrete time, decomposition methods, optimization problems

1. INTRODUCTION

There is a great incentive for developing systematic approaches that effectively identify strategies for planning and designing oilfield complexes, due to the economic impact of the underlying decisions. On the other hand, the application of optimization techniques in problems that involve oilfield exploration represents a challenging and complex problem.

The literature presents models and solution techniques for solving problems in the design and planning of infrastructure in oilfields. This problem has been initially presented in the literature by Devine and Lesso (1972) that developed an optimization model for the development of offshore oilfields.

According to Van den Heever and Grossmann (2000), in the past decisions that concerned platform capacities, scheduling of perforations and production yields had been frequently made separately. Moreover, certain assumptions were made in order to reduce the required computational effort. Another approach was to assume a fixed perforation schedule and then to determine the production yield from an LP model. A third approach was to determine the perforation schedule for a fixed production yield from an LP and subsequently round the non integer solution to integer values or even to solve the MILP.

Frair (1973) proposed independent models for calculating the number of production platforms, their capacities and the scheduling of well perforation. However, this approach has lead to infeasible or sub-optimal decisions since these were not considered in an integrated model.

Iyer *et al.* (1998) proposed a multiperiod MINLP for the planning and scheduling of investment and operation in offshore oilfields. The formulation incorporates the nonlinear behavior of the reservoirs, pressure constraints in the well surface and equipment constraints. The formulation presents a general objective function that optimizes a given economic indicator, such as NPV. A sequential decomposition technique is proposed to solve the problem that relies on the aggregation of time periods followed by successive disaggregating steps.

In the case of the planning of infrastructure of petroleum fields, MINLP models have been avoided in favor of MILP or even LP models, because of the inherent difficulties of treating nonlinear constraints and in the latter case because of the combinatorial explosion that results from discrete decisions.

Iyer and Grossmann (1998) proposed a decomposition algorithm that solves a design problem in reduced space of binary variables to determine the assignment of wells to platforms. The planning model is then solved for fixed values determined in the design subproblem.

Tsarbopoulou (2000) proposed an MILP model for the optimization of the exploration of oil and gas in a petroleum platform. The proposed model relies on binary variables to determine the existence of a given platform and the potential connection between wells and platforms.

This paper proposes a reformulation of the MILP of Tsarbopoulou (2000) model that relies on a smaller number of binary variables that requires a smaller computational effort. Moreover, a disaggregation technique proposed by Iyer and Grossmann (1998) is applied to the reformulated model that is composed of assignment and planning sub problems. The master problem determines the assignment of platforms to wells and the planning sub problem that calculates the timing for fixed assignments.

2. PROBLEM DEFINITION

An offshore oil field consists of J wells that contain oil and gas. Platforms are required to extract these substances from one or more oilfields. The planning problem involves the selection of the number and types of units, such as platforms and wells, as well as the decision of assigning platforms to wells in a given time horizon.

3. MATHEMATICAL MODEL

The planning of infrastructure in offshore oilfields includes discrete and continuous decisions along the project lifetime, such as the selection of platforms and oilfields to invest as well as oil and gas production, respectively.

Based on these considerations, the model that represents the infrastructure is a Mixed Integer Programming (MIP) problem. The objective is to maximize the net present value (NPV).

3.1 Model Assumptions

The following are the main assumptions for the proposed model:

(A1) Only two substances are removed, which are oil and gas.
(A2) The productivity index is assumed constant throughout the planning horizon.
(A3) Whenever oil is removed from a reservoir, its pressure decreases linearly.
(A4) All wells in the reservoir were connected and therefore the pressure in each well is constant in a given time period.
(A5) There is no pressure loss along the pipelines between the wells and the platforms.
(A6) A linear model represents the gas-to-oil rate. This value is 0.7 when no oil is removed and reaches the maximum value of 1.0 when all the oil is removed.
(A7) The initial amounts of each substance are known for each well.
(A8) The production limit for each substance is known along the planning horizon.
(A9) The area of the field is known and it is divided into small rectangles. In the center of each rectangle it is possible to allocate a platform.
(A10) The wells are randomly distributed in the field.
(A12) The time horizon is discretized in intervals of equal length.
(A13) Production costs and yields all substances are known for each time period.
(A14) Interest and inflation rates are known and are constant along the planning horizon.

3.2 Notation

Indices:

g	gas
i	platform
j	well
o	oil
s	substance (gas or oil)
t	time period

Continuous variables:

CON	connection cost
CUM_{st}	cumulative substance production up to year t
DR	overall drilling cost
$FMAX_{sjt}$	maximum flow of substances from well at time period t
F_{sjt}	flow rate of substance s from well j during time period t
GOR_t	gas-to-oil ratio at time period t
P_t	pressure of all wells at time period t
ZI	objective function

Binary Variables

M_i	existence of platform i
x_{ijt}	connection of platform i to well j at time period t
X_{ij}	connection of platform i to well j

Parameters:

APG_t	annual gas price at time period t
APO_t	annual oil price at time period t
$COST_{ij}$	connection cost
D_t	depreciation rate at time period t
$INVAL_{sj}$	initial value for substance s in well j
Q_{st}	upper production limit for each component at time period t
PCG	production costs for gas
PCO	production costs for oil

PI_j productivity index for well j

Problem MR corresponds to a reformulation model from the one proposed by Tsarbopoulou (2000) denoted as model MO. The main difference between both models relies on the representation of the decision variables. Tsarbopoulou (2000) considered an extra binary variable that assigns wells to platforms besides the one that relates wells to platforms at every time period (x_{ijt}). The reformulated model contains only the last set of variables, which is sufficient to model the discrete decisions of the problem.

MR:

Max Zi=GAS+OIL-DR-CON (1)

s.t.

$$CUM_{s,t} = CUM_{s,t-1} + \sum_j F_{s,j,t} \quad \forall s,t \tag{2}$$

$$GOR_t = 0.7 + 3.10^{-8} \times CUM_{o,t} \quad \forall t \tag{3}$$

$$P_t = 100 - 0.000008 \times CUM_{o,t} \quad \forall t \tag{4}$$

$$FMAX_{o,j,t} = PI_j \times P_t \quad \forall j,t \tag{5}$$

$$FMAX_{g,j,t} = PI_j (60 - 2.6.10^{-6} CUM_{o,t}) \quad \forall j,t \tag{6}$$

$$F_{s,j,t} \le FMAX_{s,j,t} \quad \forall s, j, t \tag{7}$$

$$\sum_j F_{s,j,t} \le Q_{s,t} \quad \forall s, t \tag{8}$$

$$\sum_t F_{s,j,t} \le INVAL_{s,j} \quad \forall s, j \tag{9}$$

$$DR = \sum_i (100 M_i + 10 \times \sum_j \sum_t x_{i,j,t})10000 \tag{10}$$

$$CON = \sum_i \sum_j \sum_t COST_{i,j} \times x_{i,j,t} \tag{11}$$

$$A_{j,t} = A_{j,t-1} + \sum_i x_{i,j,t} \quad \forall j,t \tag{12}$$

$$F_{o,j,t} \le FOMAX \times A_{j,t} \quad \forall j,t \tag{13}$$

$$F_{g,j,t} \le FGMAX \times A_{j,t} \quad \forall j,t \tag{14}$$

$$\sum_i \sum_t x_{i,j,t} \le 1 \quad \forall j \tag{15}$$

$$\sum_t x_{ijt} \le M_i \quad \forall i, j \tag{16}$$

$$OIL = \sum_i \sum_j \sum_t \left[F_{o,j,t} \times (APO_t - PCO) \times D_t \right] \tag{17}$$

$$GAS = \sum_i \sum_j \sum_t \left[F_{g,j,t} \times (APG_t - PCG) \times D_t \right] \tag{18}$$

The objective function in eq. 1 is the expected net present value, which includes the revenues of oil and gas reduced by the drilling and connection costs. Equation 2 states that the cumulative production of each substance (oil/gas) is the same as the cumulative production in the previous time period increased by an amount equal to the flow from all wells at the present time. Equation 3 states that the gas-to-oil rate increases as oil is extracted. Equation 4 states that the initial pressure of the reservoir is 100 bar and that it decreases linearly with accumulated production. Equations 5 and 6 are related with the maximum flow of production of the oil and gas, respectively. Equation 7 states that the flows of each substance from each well should not exceed the maximum production limits. Equation 8 states that the flow any substance, from all the wells, should not exceed the upper production limits. Equation 9 states that the flow of all substances throughout the time horizon should not exceed their initial amounts. Equations 10 and 11 are related to the drilling and connection costs, respectively. The cost depends directly on the assignment of the well to the platform at time period t. Equation 12 states that a well is opened only once and remains open throughout the whole time period. Equations 13 and 14 state that the oil and gas flow should not exceed some specific limits. Equation 15 states that a well is connected to a platform once. Equation 16 states that a well is connected to a platform only if the same platform was allocated. Equations 17 and 18 are related to the revenues from oil and gas sales, respectively.

4. DISAGGREGATION APPROACH

Iyer and Grossmann (1998) proposed a two-level decomposition approach for the planning of process networks. Van der Heever and Grossmann (2000) then applied this approach to an oilfield infrastructure-planning model. In this section, a similar approach is applied to the reformulated model MR. The disaggregated model is denoted as MD that is decomposed into two subproblems: the master subproblem that solves a model that assigns platforms to wells (problem AP) and the timing subproblem (problem TP). The latter relies on the assignments that are obtained in the master subproblem and decides on when to install the platforms. The decomposition algorithm as applied to model MR can be seen in Figure 1. The proposed technique is similar to the one proposed by Van der Heever and Grossmann (2000), which however have considered non convex nonlinearities in the sub-problem and therefore could not guarantee global solutions.

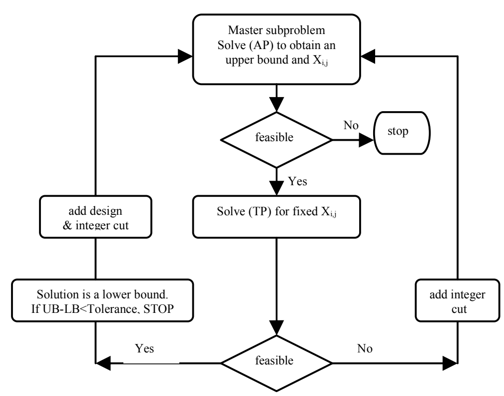

Fig. 1. Bilevel decomposition algorithm.

The assignment problem (AP) is defined as follows:

$$\max \quad ZI = GAS + OIL - DR - CON \tag{1}$$

s.t.

constraints (1) to (9), (17) and (18)

$$DR = \sum_i (100 M_i + 10 \sum_j X_{i,j})10000 \tag{19}$$

$$CON = \sum_i \sum_j \sum_t COST_{i,j} \times X_{i,j} \tag{20}$$

$$A_j = \sum_i X_{i,j} \quad \forall j \tag{21}$$

$$F_{o,j,t} \le FOMAX \times A_j \qquad \forall j,t \tag{22}$$

$$F_{g,j,t} \le FGMAX \times A_j \qquad \forall j,t \tag{23}$$

$$\sum_i X_{i,j} \le 1 \qquad \forall j \tag{24}$$

$$X_{ij} \le M_i \qquad \forall i,j \tag{25}$$

The solution of AP provides values for $X_{i,j}$. If this variable is fixed ($\overline{X}_{i,j}$), a feasible solution for TP is a feasible solution for MR and generates a lower bound for this problem, where TP is defined as follows:

Problem TP

$$\max \quad ZI = GAS + OIL - DR - CON \tag{1}$$

s.t. constraints (2) to (18)

$$x_{i,j,t} \le \overline{X}_{i,j} \qquad \forall i,j,t \tag{26}$$

$$A_{j,t} \le A_j \qquad \forall j,t \tag{27}$$

Similarly to Iyer and Grossmann (1998), constraints 26 and 27 select a subset of assignments for the planning problem.

The following are the constraints used in the algorithm to avoid subsets and supersets that would result in suboptimal solutions:

$$\sum_{n1\in Z_1^r}\sum_{n2\in Z_1^r} X_{n1,n2} + X_{i,j} \le \left|Z_1^r\right| \quad \forall i,j \in Z_0^r, r=1...R \tag{28}$$

$$\sum_{n1\in Z_0^r}\sum_{n2\in Z_0^r} X_{n1,n2} + X_{i,j} \ge 1 \quad \forall i,j \in Z_1^r, r=1...R \tag{29}$$

$$\sum_{i\in M_r}\sum_{j\in M_r} X^r_{i,j} - \sum_{i\in N_r}\sum_{j\in N_r} X^r_{i,j} \le \left|M_r\right| - 1 \tag{30}$$

where

$$M_r = \left\{ i \,/\, \overline{X}^r_{i,j} = 1 \text{ for configuration in iteration } r \right\}$$

$$N_r = \left\{ i \,/\, \overline{X}^r_{i,j} = 0 \text{ for configuration in iteration } r \right\}$$

$$Z_1^r = \left\{ i,j \,/\, X_{i,j} = 1 \right\}$$

$$Z_0^r = \left\{ i,j \,/\, X_{i,j} = 0 \right\}$$

Similarly to Iyer and Grossmann (1998), equation 28 states that if in any solution all the x variables in any set Z_1^r are 1, then all remaining variables must be zero in order to prevent a superset of Z_1^r from entering the solution of AP. Equation 29 shows cuts for precluding subsets of Z_1^r. Equation 30 has the effect of establishing the basis for deriving integer cuts on supersets and subsets of the configurations predicted by the assignment problem.

5. RESULTS

In this section, problems are solved to illustrate the performance of the model and of the solution strategy. The problems were modeled using GAMS (Brooke et al. (1998) and solved in the full space using the CPLEX solver (ILOG, 1999).

The reformulated model (MR) presented better computational performance with respect to the original model (MO) proposed by Tsarbopoulou (2000), as shown in Table 1 that presents he CPU times obtained for a problem with 16 platforms as a function of the number of wells (NW). Interestingly, the integrality gap is the same for both models and increases with problems size.

Note from Table 1 that none of the models is able to solve problems for more than 40 wells, despite a relatively small integrality gap verified for the smaller instances. Nevertheless, when MR is subject to the disaggregation strategy proposed in the previous section (denoted as MD), the computational gain is remarkable. The CPU times obtained for a problem with 16 platforms as a function of NW are compared to those from MO and MR in Figure 2.

Figures 3 and 4 illustrate the computational time for MD for different numbers of wells and for platforms, ranging from16 to 64.

Table 1 – Computational performance of the models

NW	CPU time (s) MO	CPU time (s) MR	gap (%)
05	0.8	0. 7	0.58
10	2.7	1.6	0.55
15	8.5	4.0	0.67
20	54.6	43.8	0.90
25	117.0	40.2	1.13
30	6.1	3.9	1.29
35	10.7	6.4	1.35
40	*	*	-

* No solution obtained after 18,000 CPU s.

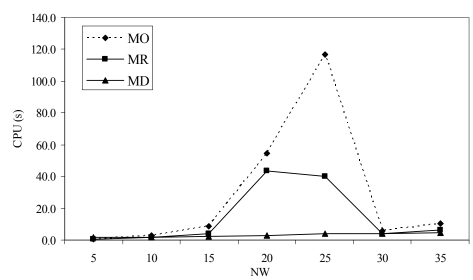

Fig. 2. CPU times for the proposed models

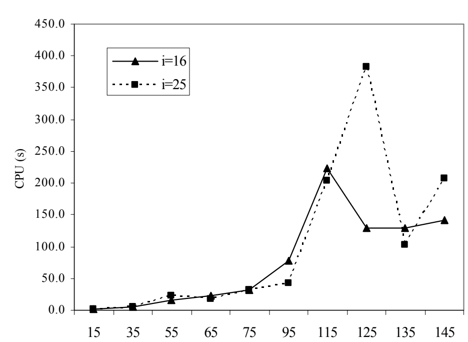

Fig. 3. Computational performance for MD in large instances

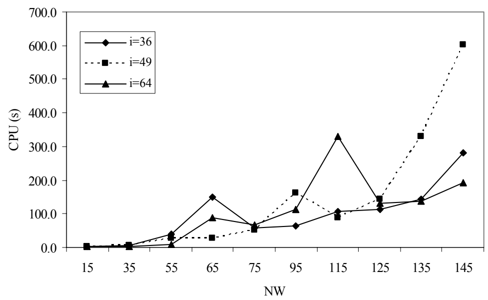

Fig. 4. Computational performance for MD in large instances

Table 2 presents the sizes of problems MO and MR, such as the number of equations (SE), number of continuous variables (SV) and number of discrete variables (DV) for several numbers of wells (NW) and 16 platforms.

Table 3 presents the corresponding sizes of problem MD, for several values of the number of wells. At each iteration, SV and DV are maintained, whereas there is an average increase de 20% in the number of equations from iteration 1 to 2, due to cut generation.

It can be seen from Table 3 that the reduction in the number of discrete values (DV) in MD is not significant with respect to MR. However the introduction of constraints (26) greatly reduces the search space and therefore the computational effort.

Table 2 – Dimensions of MO and MR

NW	MO			MR		
	SE	SV	DV	SE	SV	DV
05	1506	1351	1056	510	1111	816
10	2931	2481	1936	955	2161	1616
15	4356	3611	2816	1400	3211	2416
20	5781	4741	3696	1845	4261	3216
25	7206	5871	4576	2290	5311	4016
30	8631	7001	5456	2735	6361	4816
35	10056	8131	6336	3180	7411	5616

Table 3 – Size of problem MD

NW	Sub problem	1st iteration		
		SE	SV	DV
5	AP	465	346	101
	TP	1285	1111	866
10	AP	865	631	186
	TP	2505	2161	1716
15	AP	1265	916	271
	TP	3725	3211	2566
20	AP	1665	1201	356
	TP	4945	4261	3416
25	AP	2065	1486	441
	TP	6057	5311	4266
30	AP	2465	1771	526
	TP	7277	6361	5116
35	AP	2865	2056	611
	TP	8551	7411	5966

6. CASE STUDY

In this section we present in detail a case study as the one presented by Tsarbopoulou (2000) that provides a comparison between MO developed by the author and the proposed model MD. For this case 16 platforms and 30 wells are considered for a horizon of 10 years. In this problem, a rectangular oilfield of 10,000 ft by 15,000 ft was assumed. The interest rate was set to 15% and annual inflation rate to 3%. Upper production limits of oil and gas in each well are 1,250,000 and 875,000, respectively.

Data regarding productivity indexes (PI), initial amount of substances (oil and gas), the coordinates in the field, and depth (DP) in each well are given in Table 4. Cost and depreciation correlations, that depend on the well depth, as well as gas and oil prices are given in Tsarbopoulou (2000).

The optimal values obtained with MO, MR and MD are the same and reach $1.6464*10^9$. However, as can be noted from Table 1 and Figure 2, there is a reduction of approximately 60% in CPU time for MD. Only 3 subproblems are required for MD.

The well-platform assignments obtained for all iterations of MD are given in Table 5. Note that the sequence for the decision variable is (well, platform, time period).

Table 4 - Data for each well

j	X (ft)	Y (ft)	DP (ft)	PI $\left(\frac{ft^3}{yr.bar}\right)$	INVAL (10^5 ft^3/year)	
					Oil	Gas
1	5336	1183	6.27	1840	8.5	5.95
2	6136	4283	5.26	2000	11.0	7.70
3	6338	6640	5.34	1760	12.0	8.40
4	12911	1082	5.61	1920	9.5	6.65
5	4528	8700	5.92	1980	10.0	7.00
6	10862	8990	5.16	1680	10.5	7.35
7	9683	4679	5.42	1620	8.0	5.60
8	2716	2677	5.11	1629	9.0	6.30
9	8808	4510	5.82	1740	10.0	7.00
10	6007	5702	5.66	1940	11.5	8.05
11	2999	6058	5.00	1840	8.5	5.95
12	13090	2313	6.22	2000	11.0	7.70
13	13855	5889	6.25	1760	12.0	8.40
14	7713	6440	4.90	1920	9.5	6.65
15	4369	2773	5.59	1980	10.0	7.00
16	10260	8099	5.26	1680	10.5	7.35
17	11416	4973	6.03	1620	8.0	5.60
18	6648	3866	5.17	1629	9.0	6.30
19	9834	3451	5.57	1740	10.0	7.00
20	8006	3679	5.73	1940	11.5	8.05
21	12096	2913	4.88	1840	8.5	5.95
22	7000	7869	4.58	2000	11.0	7.70
23	3477	1774	5.78	1760	12.0	8.40
24	9153	3104	6.08	1920	9.5	6.65
25	617	1034	4.76	1980	10.0	7.00
26	1071	3328	5.06	1680	10.5	7.35
27	4095	1249	5.06	1620	8.0	5.60
28	7440	9979	5.98	1629	9.0	6.30
29	7155	9232	6.29	1740	10.0	7.00
30	1095	7980	6.36	1940	11.5	8.05

Table 5 – Assignments for each iteration of MD

r=1		r=2
AP - $X_{i,j}^{(1)}$	TP - $x_{i,j,t}^{(1)}$	AP - $X_{i,j}^{(2)}$
2, 3	2, 3, 1	2, 3
2, 6	2, 6, 1	2, 6
2, 7	2, 7, 1	2, 7
2, 11	2, 11, 1	2, 11
2, 14	2, 14, 1	2, 14
2, 23	2, 23, 1	2, 23
3, 12	3, 12, 1	3, 12
3, 13	3, 13, 1	3, 13
3, 15	3, 15, 1	3, 15
3, 19	3, 19, 1	3, 19
4, 4	4, 4, 1	4, 4
4, 10	4, 10, 1	4, 10
9, 1	9, 1, 1	9, 1
9, 8	9, 8, 1	9, 8
9, 22	9, 22 ,1	9, 22
9, 24	9, 24, 1	9, 24
9, 29	9, 29, 1	9, 29
9, 30	9, 30, 1	9, 30
10, 2	10, 2, 1	10, 2
10, 5	10, 5, 1	10, 5
10, 9	10, 9, 1	10, 20
10, 20	10, 20, 1	11, 9
11, 25	11, 25, 1	11, 25
11, 28	11, 28, 1	11, 28
13, 18	13, 18, 1	13, 18
13, 27	13, 27, 1	13, 27
15, 17	15, 17, 1	15, 17
15, 21	15, 21, 1	15, 21

Note that constraints 28 to 30 do not allow the repetition of assignments neither the generation of sub and supersets. In this sense there is no significant change in the allocation obtained in AP in consecutive iterations. The only modification is the allocation of well 9 to platform 11 in iteration 2 in place of the assignment of well 20 to platform 10.

The cumulative productions of oil and gas as well as GOR, as functions of time are shown in Figures 5 and 6, respectively. Note that the cumulative flow rate of oil as well as the gas to oil ratio increase linearly with time up to time period 8. Afterwards, there is a reduction in this value due to the upper bound on the GOR that is set to one.

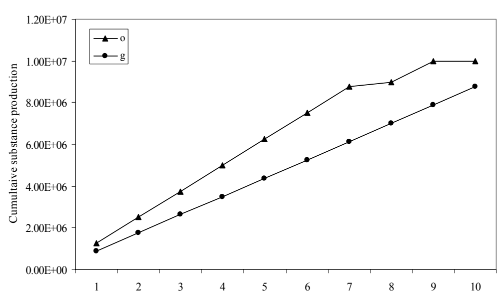

Fig. 5. Cumulative substance production up to year t

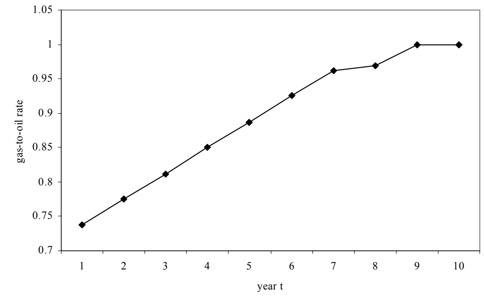

Fig. 6. Gas-to-oil ratio at year t

7. CONCLUSIONS

This paper presented a reformulated MILP for the planning of the oilfield infrastructure that presents a significant reduction in the number of discrete variables for the same relaxation gap with respect to the model developed by Tsarbopoulou (2000). Moreover, a decomposition approach that relies on the disaggregation of the assignment and timing decisions in analogy to the one proposed by Iyer and Grossmann (1998) has been presented. Results show that computational performance is greatly improved, whereas global optimality is guaranteed. Problems of 64 platforms and 145 wells are efficiently solved for a 10-year horizon.

ACKNOWLEDGMENT - The authors acknowledge financial support from CAPES (Brazil).

REFERENCES

Brooke, A., Kendrick, D. and Meeraus, A. (1998). GAMS: A User´s Guide; The Scientific Press: San Francisco, CA.

Devine, M.D. and Lesso, W.G. (1972). Models for the minimum cost development of offshore oil fields. In: *Manage. Sc.* Vol. 18. Nº. 8, B378-B387.

Ilog (1999). Ilog Cplex (6.5) User´s Manual. Ilog Corp. Gentilly (France):

Iyer, R.R. and Grossmann, I.E. (1998). A bilevel decomposition algorithm for long-range planning of process networks. In: *Ind. Eng. Chem. Res*, Vol. 37, 474-481.

Iyer, R.R., Grossmann, I.E., Vasantharajan, S. and Cullick, A.S. (1998) Optimal planning and scheduling of offshore oil field infrastructure investment and operations. *Ind. Eng. Chem. Res*, Vol. 37, 1380-1397.

Tsarbopoulou, C. (2000). Optimisation of oil facilities and oil production, M.Sc. Dissertation, UCL, London, UK.

Van den Heever, S., Grossmann, I.E., Vasantharajan, S. and Edwards, K. (2000). Integrating complex economic objectives with the design and planning of offshore oilfield infrastructures. *Comput. Chem. Engng*, Vol. 24, 1049-1055.

Van den Heever, S. and Grossmann, I.E. (2000). An Iterative Aggregation / Disaggregation Approach for the Solution of a Mixed-Integer Nonlinear Oilfield Infrastructure Planning Model. *Ind. Eng. Chem. Res*, Vol. 39, 1955-1971.