Korean J. Chem. Eng., **22**(2), 173-183 (2005)

Multiloop PID Controller Design using Partial Least Squares Decoupling Structure

Junghui Chen†, Yi-Chun Cheng and Yuezhi Yea

Department of Chemical Engineering, Chung-Yuan Christian University, Chung-Li, Taiwan 320, Republic of China
(*Received 12 June 2004 • accepted 6 November 2004*)

Abstract–The goal of this paper is to identify and control multi-input multi-output (MIMO) processes by means of the dynamic partial least squares (PLS) model, which consists of a memoryless PLS model connected in series with linear dynamic models. Unlike the traditional decoupling MIMO process, the dynamic PLS model can decompose the MIMO process into a multiloop control system in a reduced subspace. Without the decoupler design, the optimal tuning multiloop PID controller based on the concept of general minimum variance and the constrained criteria can be directly and separately applied to each control loop under the proposed PLS modeling structure. Several potential applications using this technique are demonstrated.

Key words: Partial Least Squares, Multiloop, PID Controllers, Adaptive Control, Multivariable Process

INTRODUCTION

Controlled processes in nearly all chemical industries frequently encounter the need for more than one variable to be controlled. They are known as multivariable or multi-input multi-output (MIMO) processes. The control of multivariable systems is not always an easy task due to its complex and interactive nature. Most of the automatic tuning methods are intended for single-input single-output (SISO) processes. Few of them are intended for MIMO processes [Kashiwagi and Li, 2004; Palmor et al., 1995; Oh and Yeo, 1995; Zhuang and Atherton, 1994] because SISO is easy to understand and readily available in hardware and software. A critical step in the multiloop dynamic control design of the dynamic MIMO is the development of a suitable model that pairs the control loops. Multivariable PID controllers were discussed in some previous research. The biggest log modulus tuning (BLT) method [Luyben, 1986] was designed for each loop, respectively, depending on the tradeoff between stability and performance of the system. The controller parameters would be properly adjusted by a detuning factor to maintain the stability. The internal model control [Garcia and Morari, 1982] approach was more amenable, but it required full knowledge of the process. Recently, a sequential design was used for multiloop PID controller systems [Shiu and Huang, 1998]. It usually took a great deal of time to identify the multiloop system and design the multi single-loop controller in a sequential procedure.

The development of chemometric techniques has spurred a torrent of research in multivariable processes. Those techniques can be used to extract the state of the system via applications of mathematical and statistical methods from the stored data. Several chemometric techniques were proposed, like principal component analysis and partial least squares (PLS). They have received considerable attention in the field of chemical process problems and have been applied to system monitoring and diagnosis [Han et al., 2003; Chen and Yen, 2003; Kourti and MacGregor, 1996; Ku et al., 1995]. Still, it was rarely on the control problem. All of these works showed that only a few principal components could capture most of the characteristics of the system pattern in a multivariable process behavior. They also could successfully tackle operational data analysis. Recently, dynamic PLS for control system design was addressed [Kaspar and Ray, 1992, 1993]. Dynamic PLS incorporated a dynamic transformation into the standard PLS model. Then the synthesis method of controller design was used to tune the controller parameters on each of the control loops. Another dynamic PLS approach, also known as a projection-based dynamic model, was proposed [Lakshminaraynan et al., 1997]. A PLS outer model was first constructed. The dynamic relationship between the input and the output scores was built on the inner model. However, the control objective was still lumped when model predictive control was used. Although the control performances of the above methods were satisfactory, the adaptive control design based on the PLS related models has not been developed.

In the control design, it might be difficult to deal with a time varying chemical process. To improve the control performance, several schemes of self-tuning PID controllers were proposed in the past. Wittenmark [1979] developed the control structure with the PID algorithm calculated via pole placement design. The method was limited in the order of the controlled process. The self-tuning PI or PID algorithms were automatically derived from the dynamic of the controlled processes [Gawthrop, 1986]. An alternative self-tuning PID controller was built based on the generalized minimum variance control [Cameron and Seborg, 1983]. The control structure was oriented to have a PID structure. The controller parameters were obtained by using a parameter estimation scheme. The PID adaptive algorithm with the combination of traditional control design methods (Ziegler-Nichols method and pole placement design) and a recursive identification procedure was also developed [Bohim Bobal and Prokop, 1999; Banyasz and Keviczky, 1993]. Other forms of self-tuning PID can be found in literature [Radke and Isermann, 1987; Ortega and Kelly, 1984; Proudfoot et al., 1983]. However, the above self-tuning adaptive control approaches are all useful only for the SISO system.

To improve the multiloop control performance, an adaptive multiloop PID algorithm is proposed. It combines the general minimum

†To whom correspondence should be addressed.
E-mail: jason@wavenet.cycu.edu.tw

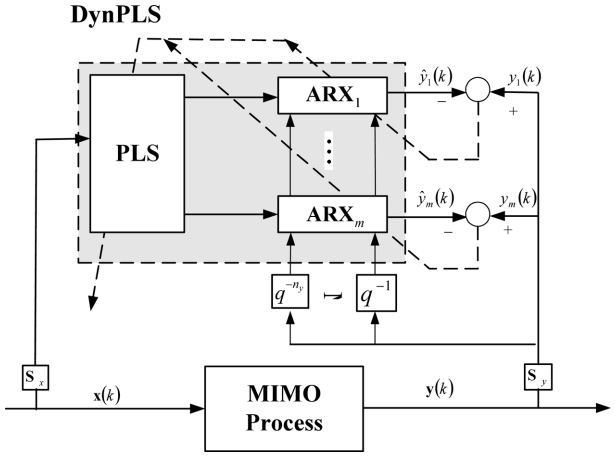


Fig. 1. Structure of the generalized MIMO DynPLS model. S_x and S_y are the factors that scale the input and output variables, respectively.

variance (GMV) with the decomposition characteristic of the dynamic PLS structure. Dynamic PLS consists of the cascade connection of a memoryless PLS gain matrix followed by linear dynamic models. With the dynamic PLS model, the PID algorithm can be implemented directly onto each control loop without any modification.

STRUCTURE OF DYNAMIC PLS MODEL

With M controlled and N manipulated variables, an MIMO process whose number of inputs and outputs may be unequal is given in Fig. 1. There is often coupling in this process; that is, a large disturbance from other loops occurs whenever the manipulated variable of one loop changes. This interaction may cause oscillation and even instability. In this section, the PLS technique is used to eliminate the interaction of the MIMO system. First, a brief overview of the multivariate PLS statistical technique is presented. Then the controlled and manipulated variables can be transformed into a smaller informative set via a set of linear functions which model the combinational relationship between the controlled variables and latent controlled variables, and between the manipulated variables and latent manipulated variables, respectively. Finally, the static PLS is extended to dynamic PLS to handle the process measurements with the dynamic behavior.

1. Partial Least Squares

PLS regression derived from the classical linear regression is often used to predict properties of processes based on variables only indirectly related to the properties. The given process data are subdivided into two blocks, a dependent block (**Y**) and an independent block (**X**). **Y** block with a two-way array (I×M) summarizes the I samples and the M final quality variables. **X** block with a two-way array (I×N) organizes process operating N variables. PLS is used to extract latent variables. The latent variables explain the best correlation between the product quality block (**Y**) and the process data block (**X**).

The standard PLS regression [Höskuldsson, 1988] relies on decomposing the dependent block (**Y**) and the independent block (**X**) into a sum of rank one component matrices. Before PLS is applied, each measurement variable that centers and scales the variance to unit one is typically applied; this will put all variables on an equal basis. Initially, let $\mathbf{X}_0=\mathbf{X}$, $\mathbf{Y}_0=\mathbf{Y}$ and r=0. Find a vector (or component) ($\mathbf{w}_r$) which is correlated with $\mathbf{Y}_r$ while describing a large amount of the variation in $\mathbf{X}_r$. It can be formulated as

$$\mathbf{w}_r = \arg\max_{\mathbf{w}_r}(\mathbf{X}_r^T\mathbf{Y}_r, \|\mathbf{w}_r\|=1) \tag{1}$$

Then the component is subtracted from $\mathbf{X}_r$ and $\mathbf{Y}_r$

$$\mathbf{X}_r=\mathbf{X}_{r-1}-\mathbf{t}_r\mathbf{w}_r^T \tag{2}$$

$$\mathbf{Y}_r=\mathbf{Y}_{r-1}-b_r\mathbf{t}_r\mathbf{c}_r^T \tag{3}$$

and

$$\mathbf{t}_r=\mathbf{X}_r\mathbf{w}_r \tag{4}$$

$$\mathbf{c}_r=\frac{\mathbf{Y}_r^T\mathbf{t}_r}{\mathbf{t}_r^T\mathbf{t}_r},\ \mathbf{u}_r=\mathbf{Y}_r^T\mathbf{c}_r \tag{5}$$

where $\mathbf{w}_r$ and $\mathbf{c}_r$ are the loadings of $\mathbf{X}_r$ and $\mathbf{Y}_r$, respectively. The score ($\mathbf{t}_r$) is the projection of $\mathbf{X}_r$ into the direction $\mathbf{w}_r$. The score ($\mathbf{u}_r$) is the projection of $\mathbf{Y}_r^T$ into the direction $\mathbf{c}_r$. b_r is the regression coefficient related to $\mathbf{t}_r$ and $\mathbf{u}_r$,

$$b_r=\frac{\mathbf{u}_r^T\mathbf{t}_r}{\mathbf{t}_r^T\mathbf{t}_r} \tag{6}$$

Eqs. (2) and (3) are used to remove the variance associated with the already calculated *r*-th directions of $\mathbf{w}_r$ and $\mathbf{c}_r$ in the variance of process variables and quality variables, respectively. Then set r=r+1 and repeat the above procedures (Eqs. (1)-(6)) until the description of **Y** convergence is properly gotten. Finally, the matrices **Y** and **X** are separately decomposed into the summation of the product of score vectors **t** and loading vectors **w** and **c** plus some residual matrix **E** and **F**, respectively:

$$\mathbf{X}=\sum_{r=1}^{R}\mathbf{t}_r\mathbf{w}_r^T+\mathbf{E}=\mathbf{T}\mathbf{W}^T+\mathbf{E}$$

$$\mathbf{Y}=\sum_{r=1}^{R}\mathbf{t}_r\mathbf{c}_r^T+\mathbf{F}=\mathbf{T}\mathbf{C}^T+\mathbf{F} \tag{7}$$

where R is the number of principal components retained in PLS. Due to its simplicity and easy interpretation, the applications of this approach can be found in an abundant literature. However, the PLS model only deals with static rather than dynamic relationships. This means that it is in the form of a trace of processing without memory in the previous time observations. It will limit its use to a typical off-line atmosphere since it does not comply with the dynamic controlled process that contains the serial correlation among the process variables.

2. Dynamic PLS Model

A simple method to modify PLS for handling the autocorrelation data is to mimic the concept of the auto-regressive exogenous time series model by forming the data matrix with the previous observations in each observation vector [Qin and McAvoy, 1992]. Ricker [1988] applied the finite impulse response of the process variable to PLS. However, the above methods need to substantially increase the dimensions of the input and output matrices. Kaspar and Ray [1992] proposed the standard PLS procedure should be used after the dynamic component input data were filtered. The dynamic filter is restricted to the first-order plus the dead time, computed by a prior

knowledge of the process [Kaspar and Ray, 1992] or by a minimum criterion of the prediction residuals [Kaspar and Ray, 1993].

In this research paper, ARX is integrated with PLS. This proposed combinational method, referred to as Dynamic PLS (DynPLS), can extract the time-dependent relations in the measurements. DynPLS consists of a cascade connection of a memoryless PLS model and linear dynamic ARXs (Fig. 1). PLS with multiple inputs and outputs can be modeled for multivariable static systems. The dynamic characteristics of the system can often be inferred from the analysis of a time series ARX model fitted to the observations of the system. The MIMO model can be expressed as:

$$\mathbf{y}^h(k)=\mathrm{PLS}(\mathbf{x}(k))$$
$$\hat{\mathbf{y}}(k+1)=\sum_{i=0}^{n_a-1}\mathbf{A}_i\mathbf{y}(k-i)+\mathbf{y}^s(k) \quad (8)$$

where $\mathbf{y}^h(k)=[y_1^h(k)\ \cdots\ y_M^h(k)]^T$ is the output vector from PLS, $\mathbf{y}(k)=[y_1(k)\ \cdots\ y_M(k)]^T$ is the output vector, $\hat{\mathbf{y}}(k+1)$ is the one-step ahead predictor, n_a indicates the number of output lag terms, and $\mathbf{A}_i=\mathrm{diag}(a_{m,i})_{m=1,2,\ldots,M}$ is a diagnoanl matrix. $\mathbf{y}^s(k)=[y_1^h(k-d_1)\ \cdots\ y_M^h(k-d_M)]^T$ and d_i is the dead time of y_i^h. In Eq. (8), the current output value of time series is expressed as a weighted sum of the past output values plus the outputs from the PLS model. Thus, $\mathbf{y}(k+1)$ can be regressed on the n_a previous values of $\mathbf{y}(k-i)$ and the past PLS output values, $\mathbf{y}^s(k)$.

3. Identification of DynPLS Model

From the dynamic data, the steady-state PLS and the linear dynamics should be identified simultaneously. This algorithm is based on the standard alternating optimization procedure which works as follows. It begins with the chosen parameters $\mathbf{A}_i$, i=1, 2, …, n_a of the linear dynamic model. Then the parameters ($\mathbf{W}$ and $\mathbf{C}$) of PLS can be estimated by using Eqs. (1)-(6),

$$[\mathbf{W}\ \ \mathbf{C}]=\arg\min_{W,C}\|\mathbf{Y}^h-\mathrm{PLS}(\mathbf{X})\| \quad (9)$$

where $\mathbf{Y}^h=\lfloor \mathbf{y}^h(1)\ \mathbf{y}^h(2)\ \cdots\ \mathbf{y}^h(I-1-d)\rfloor$ and $d=\max_{W,C}\{d_i\}$ can be computed by substituting the observations into the ARX model of Eq. (8). With the given PLS model, the input to the linear dynamic part can be computed as:

$$\hat{y}_m(k+1)=\sum_{i=0}^{n_{ma}-1}a_{m,i}y_m(k-i)+y_m^h(k-d_m) \quad (10)$$

The parameters of the linear dynamic model, $\boldsymbol{\theta}_m=[a_{m,1}\ a_{m,2}\ \cdots\ a_{m,n_{ma}}]^T$, for the output m can be estimated by solving the following regression:

$$\mathbf{y}_m=\boldsymbol{\Phi}_m\boldsymbol{\theta}_m+\boldsymbol{\varepsilon} \quad (11)$$

where $\mathbf{y}_m=[y_m(d_m+1)-y_m^h(1)\ y_m(d_m+2)-y_m^h(2)\ \cdots\ y_m(I)-y_m^h(I-1-d_m)]^T$
$\boldsymbol{\Phi}_m=[\boldsymbol{\varphi}_m(1)\ \boldsymbol{\varphi}_m(2)\ \cdots\ \boldsymbol{\varphi}_m(I-1-d_m)]^T$ and
$\boldsymbol{\varphi}_m(k)=[y_m(d_m)\ y_m(d_m-1)\ \cdots\ y_m(d_m-n_{ma})]^T$

The least-squares estimate of the linear parameters is found through:

$$\boldsymbol{\theta}_m=[\boldsymbol{\Phi}_m^T\boldsymbol{\Phi}_m]^{-1}\boldsymbol{\Phi}_m^T\mathbf{y}_m \quad (12)$$

Then the PLS parameters are estimated again by using Eq. (9), and the whole procedure is iteratively done until the parameter difference between two successive iterations is smaller than a predefined threshold. Since it is conducted off-line, the whole algorithm must be restarted when new input-output data become available. Note that before the parameters of DynPLS model are identified, the dead time and lag orders of the ARX model should be defined first. Selecting these terms, however, is critical. Choosing the wrong lag terms or the dead time used as regressors may have a disasterous impact on some control applications. Lag terms that are too small obviously imply that the essential dynamic would not be modelled, but too large lag terms can also lead to difficulties in some of the control design. To find the correct lag orders, a stepwise model-building algorithm for estimating lag terms and dead time is employed [Chen and Yea, 2002]. Compared with Kaspar's work [1993], the proposed method shares the same feature in conducting the PLS without increasing the dimensions of inputs or outputs. Also, the dynamic model structure of the proposed method is so fexible that it can meet the process dynamic behavior. Without the nonlinear least squares method, the sequential training procedure identifies PLS and ARX separately. Not only can it decrease the dimension of the search space, but also substantially cut down the convergence time in general.

MULTILOOP PID CONTROLLER DESIGN

The block diagram of the multiloop control system to be considered is shown in Fig. 2. The MIMO system model is decomposed into several pairs of the input-output score. The multi-control loop is then applied onto each pair to form a single loop control design problem. A method of incorporating the adaptive PID control into each independent control loop is developed.

1. Conventional PID Controller

The PID controller from the process variable y(t) to the control

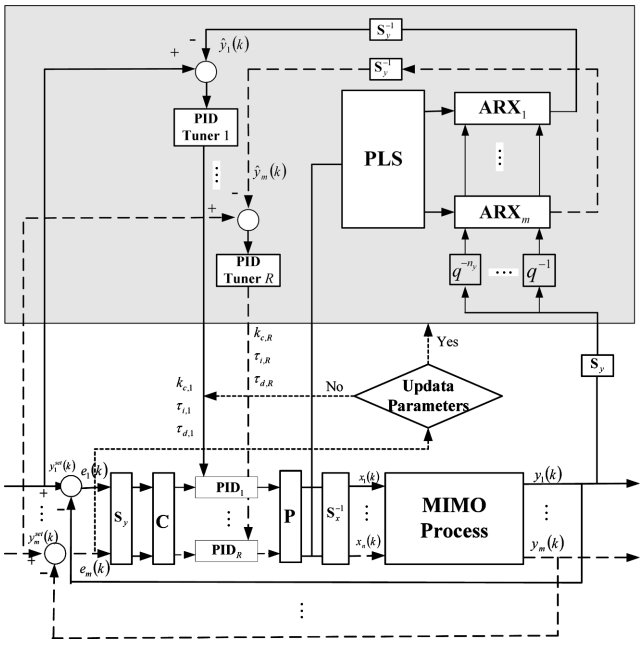


Fig. 2. Implementation of the DynPLS model-based multiloop PID controller design. S_x and S_i are the factors that scale the input and output variables, respectively. S_x^{-1} and S_y^{-1} are the factors that rescale the input and output variables, respectively.

variable x(t) is

$$x(t) = x_s + k_c\left(e(t) + \frac{1}{\tau_i}\int e(t)dt + \tau_d\frac{de(t)}{dt}\right) \quad (13)$$

where x_s is the bias value. $e(t)=y^{set}(t)-y(t)$ is the output error deviated from the setpoint. k_c, τ_i and τ_d are known as the proportional gain, the integral time constant and derivative time constant, respectively. A velocity form of the discrete PID controller whose integral action is computed by using the trapezoidal approximation can be written as

$$\Delta x(k) = x(k) - x(k-1) = k_c[(e(k) - e(k-1)) + \frac{\Delta t}{2\tau_i}(e(k)+e(k-1)) + \frac{\tau_d}{\Delta t}(e(k) - 2e(k-1) + e(k-2))] \quad (14)$$

The discrete form of the PID control is rearranged in the following form:

$$\Delta x(k) = k_0 e(k) + k_1 e(k-1) + k_2 e(k-2) = \mathbf{e}^T(k)\mathbf{k}(k) \quad (15)$$

where

$$\mathbf{e}(k) = [e(k)\ e(k-1)\ e(k-2)]^T \quad (16)$$

and

$$\mathbf{k}(k) = [k_0\ k_1\ k_2]^T = \left[k_c\left(1 + \frac{\Delta t}{2\tau_i} + \frac{\tau_d}{\Delta t}\right) - k_c\left(1 - \frac{\Delta t}{2\tau_i} + \frac{2\tau_d}{\Delta t}\right)\frac{k_c\tau_d}{\Delta t}\right]^T \quad (17)$$

2. Multiloop PID Controllers of Decoupling Structure

The goal of the controller design for the MIMO system is to seek control actions $\mathbf{x}(k)$ that can minimize the difference between the process outputs $\mathbf{y}(k)$ and the desired outputs $\mathbf{y}^{set}(k)$ at the next time step; i.e., the process outputs can reach the desired values at the next run time. Besides, from the operational point of view, the variance controller output should be minimized in order to avoid excessive control effort. The objective function of the MIMO system is expressed as

$$\min_{\substack{k_{c,r},\tau_{i,r},\tau_{d,r}\\ r=1,2,\ldots,R}} J = \frac{1}{2}\min_{\substack{k_{c,r},\tau_{i,r},\tau_{d,r}\\ r=1,2,\ldots,R}} [\|\mathbf{e}(k+1)\|^2 + \mu\|\Delta\mathbf{x}(k)\|^2] \quad (18)$$

where μ is the weighting penalty parameter. $k_{c,r}$, $\tau_{i,r}$, $\tau_{d,r}$ are the PID control parameters of the loop r. Here assume R control loops in the reduced subspace are selected. Since $\mathbf{e}(k+1)=\mathbf{y}^{set}(k+1)-\mathbf{y}(k+1)$, the objective function involves a term in the future of the next time step, namely $\mathbf{y}(k+1)$, which is not available at time k. Using Dyn-PLS model (Eq. (8)), an one-step ahead output can be predicted, that is, $\mathbf{y}(k+1) \cong \hat{\mathbf{y}}(k+1)$,

$$\min_{\substack{k_{c,r},\tau_{i,r},\tau_{d,r}\\ r=1,2,\ldots,R}} J = \frac{1}{2}\min_{\substack{k_{c,r},\tau_{i,r},\tau_{d,r}\\ r=1,2,\ldots,R}}\left[\left\|\mathbf{y}^{set}(k+1) - \left(\sum_{i=0}^{n_a-1}\mathbf{A}_i\mathbf{y}(k-i) + \mathbf{y}^s(k)\right)\right\|^2 + \mu\|\Delta\mathbf{x}(k)\|^2\right] \quad (19)$$

Let $\mathbf{y}^{s,set}(k+1) \equiv \mathbf{y}^{set}(k+1) - \sum_{i=0}^{n_a-1}\mathbf{A}_i\mathbf{y}(k-i)$, $\mathbf{y}^{s,set}(k+1)$ and $\mathbf{y}^s(k)$ can be decomposed into the lower dimensional space $\mathbf{y}^{s,set}(k+1)=\sum_{r=1}^{R} t_r^{set}(k+1)\mathbf{c}_r$ and $\mathbf{y}^s(k)=\sum_{r=1}^{R} t_r(k)\mathbf{c}_r$. The above equation can be represented as

$$J = \frac{1}{2}\min_{\substack{k_{c,r},\tau_{i,r},\tau_{d,r}\\ r=1,2,\ldots,R}}\left[\left\|\sum_{r=1}^{R}(t_r^{set}(k+1) - t_r(k))\mathbf{c}_r\right\|^2 + \mu\left\|\sum_{r=1}^{R}\Delta t_r\mathbf{p}_r^T\right\|^2\right]$$

$$\le \frac{1}{2}\min_{\substack{k_{c,r},\tau_{i,r},\tau_{d,r}\\ r=1,2,\ldots,R}}\left[\sum_{r=1}^{R}(t_r^{set}(k+1) - t_r(k))^2\|\mathbf{c}_r\|^2 + \mu\sum_{r=1}^{R}(\Delta t_r(k))^2\|\mathbf{p}_r^T\|^2\right]$$

$$= \min_{\substack{k_{c,r},\tau_{i,r},\tau_{d,r}\\ r=1,2,\ldots,R}}[J_1 + J_2 + \ldots\ldots + J_R]$$

$$= [\min_{k_{c,1},\tau_{i,1},\tau_{d,1}} J_1 + \min_{k_{c,2},\tau_{i,2},\tau_{d,2}} J_2 + \ldots\ldots + \min_{k_{c,R},\tau_{i,R},\tau_{d,R}} J_R] \quad (20)$$

This is the consequence of the Schwarz inequality. Letting $J_r \equiv 1/2[(t_r^{set}(k+1)-t_r(k))^2\|\mathbf{c}_r\|^2 + \mu(\Delta t_r(k))^2\|\mathbf{p}_r^T\|^2]$, the objective function is decomposed into R subobjective functions in the lower dimensional subspace, $J=\sum_{r=1}^{R} J_r$. Only R score variables (t_r, r=1, 2, … R) require separate design compared with M process variables to be lumped together without the decomposition. These multi-loop controllers, like decentralized controllers, have a simpler structure and, accordingly, fewer tuning parameters are needed than the fully cross-coupled one. This decomposition structure for the multidimensional control problem is a key component of the decoupling method.

3. Auto-Tuning PID Controller of Each Control Loop

After the objective function is decoupled into R objective functions, the conventional SISO controller design technique can be directly applied to each score variable, respectively, in the decomposed space, because the MIMO system is decomposed by using PLS, and the interactions which exist between control loops are also eliminated. The only difference is that the process variables are converted into the score variables in the subspace. Each subobjective (J_r) is rearranged into

$$\min_{k_{c,r},\tau_{i,r},\tau_{d,r}} J_r = \frac{1}{2}\min_{k_{c,r},\tau_{i,r},\tau_{d,r}}[(t_r^{set}(k+1) - t_r(k))^2 + \lambda_r(\Delta t_r(k))^2] \quad (21)$$

where $\|\mathbf{c}_r\|^2$ and $\|\mathbf{p}_r^T\|^2$ of Eq. (20) with the penalty factor are lumped into a coefficient λ_r. Here the incremental form of the PID controller is used in each loop,

$$t_r(k) = t_r(k-1) + \mathbf{e}_{sub,r}^T(k)\mathbf{k}_r(k) \quad (22)$$

where $\mathbf{e}_{sub,r}(k) = [e_{sub,r}(k)\ e_{sub,r}(k-1)\ e_{sub,r}(k-2)]$ and $e_{sub,r}(k) = t_r^{set}(k) - t_r(k)$

Let the change of the tuning parameter at the sampling point k be $\Delta\mathbf{k}_r(k)$, the tuning parameters $\mathbf{k}_r(k)$ at the sampling point k become

$$\mathbf{k}_r(k) = \mathbf{k}_r(k-1) + \Delta\mathbf{k}_r(k) \quad (23)$$

Substituting Eqs. (22) and (23) into the objective function gives

$$J_r = \frac{1}{2}\Delta\mathbf{k}_r^T(k)\mathbf{A}_r(k)\Delta\mathbf{k}_r(k) + \mathbf{d}_r^T(k)\Delta\mathbf{k}_r(k) + c_r \quad (24)$$

where

$$\mathbf{A}_r(k) = (\mathbf{I} + \lambda_r)\mathbf{e}_{sub,r}(k)\mathbf{e}_{sub,r}^T(k)$$
$$\mathbf{d}_r^T(k) = [-(t_r^{set}(k) - t_r(k))\mathbf{e}_{sub,r}^T(k) + (1+\lambda_r)\mathbf{k}_r^T(k-1)\mathbf{e}_{sub,r}(k)\mathbf{e}_{sub,r}^T(k)]$$
$$c_r = \frac{1}{2}(t_r^{set}(k) - t_r(k))^2 - (t_r^{set}(k) - t_r(k))[\mathbf{e}_{set,r}^T(k)\mathbf{k}_r(k-1)] + \frac{1}{2}(\mathbf{e}_{set,r}^T(k)\mathbf{k}_r(k-1))^2 + \frac{\lambda_r}{2}\mathbf{k}_r^T(k-1)\mathbf{e}_{sub,r}(k)[\mathbf{e}_{sub,r}^T(k)\mathbf{k}_r(k-1)] \quad (25)$$

When minimizing J_r with respect to $\Delta\mathbf{k}_r(k)$, we are seeking a set of PID controller parameters of the loop r in the quadratic function of this objective function. The gradient of J_r can be computed as

$$\nabla J_r(\Delta\mathbf{k}_r(k)) = \frac{\partial J_r(\Delta\mathbf{k}_r(k))}{\partial\Delta\mathbf{k}_r(k)} = \mathbf{A}_r(k)\Delta\mathbf{k}_r(k) + \mathbf{d}_r(k) \quad (26)$$

The optimal point will occur when the gradient is equal to zero. Thus, the required changes of the control parameters are

$$\Delta \mathbf{k}_r(k) = -\mathbf{A}_r^{-1}(k)\mathbf{d}_r(k) \tag{27}$$

Using Eq. (17), the corresponding PID control parameters of the controller loop r are

$$\begin{aligned} k_{c,r}(k) &= -[k_{1,r}(k) + 2k_{2,r}(k)] \\ \tau_{i,r}(k) &= \frac{-[k_{1,r}(k) + 2k_{2,r}(k)]\Delta t}{k_{0,r}(k) + k_{1,r}(k) + k_{2,r}(k)} \\ \tau_{d,r}(k) &= \frac{-k_{2,r}(k)\Delta t}{k_{1,r}(k) + 2k_{2,r}(k)} \end{aligned} \tag{28}$$

Each PID controller parameter can be computed directly by using Eqs. (27) and (28) without any difficulty, but physically it is not suitable because the manipulated inputs and controlled outputs based on the computed controller parameters may be out of the operating ranges. Typically, the constraints are defined at the minimum and maximum of $\mathbf{x}$ and $\mathbf{y}$, $\mathbf{x}^{min}=[x_1^{min}\ x_2^{min}\ \cdots\ x_N^{min}]$, $\mathbf{x}^{max}=[x_1^{max}\ x_2^{max}\ \cdots\ x_N^{max}]$, $\mathbf{y}^{min}=[y_1^{min}\ y_2^{min}\ \cdots\ y_M^{min}]$, and $\mathbf{y}^{max}=[y_1^{max}\ y_2^{max}\ \cdots\ y_M^{max}]$. Thus, the constraints placed on the input and output variables $\mathbf{x}$ and $\mathbf{y}$ at each sampling time are

$$\begin{aligned} \mathbf{x}^{min} \le \mathbf{x}(k) \le \mathbf{x}^{max} \\ \mathbf{y}^{min} \le \mathbf{y}(k) \le \mathbf{y}^{max} \end{aligned} \tag{29}$$

However, these constraint relationships cannot be transformed onto the latent space because the sub-optimum may occur [Lakshminarayanan et al., 1997]. In order to implement the decomposition strategy, a quadratic function (Eq. (21)) is still solved for each control loop, but the manipulated variables mapped back from the latent space and the corresponding controlled variables predicted from the DynPLS model should satisfy the constraints (Eq. (29)). If the estimated manipulated variables exceed the bounds, the bound values of the manipulated variables would be applied. This way, the manipulated variables and the controlled variables would be located in the feasible input constraint regions and the output constraint regions, respectively.

ILLUSTRATIVE EXAMPLES

Two case studies are used to illustrate the advantages of the proposed identification and PID control design methodology. They will be discussed separately in the sub-sections as follows.

1. Example 1: Nonsquare System

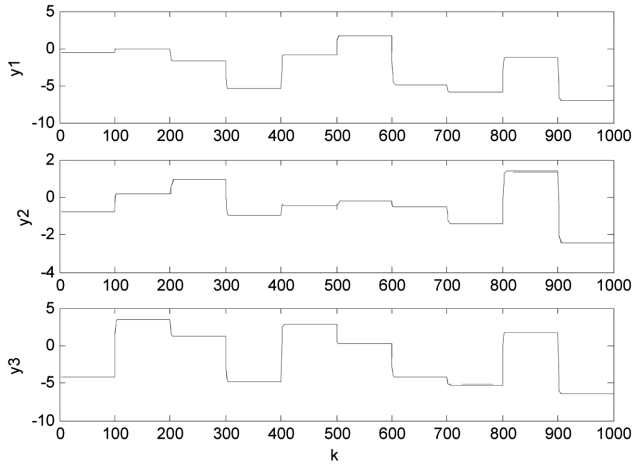
Processes with unequal number of inputs and outputs are frequently encountered in industrial processes. For the convenience of the control design, they are often squared by adding or deleting the appropriate number of inputs or outputs from the process to be controlled [Reeves and Arkun, 1989]. Here a system with four inputs and three outputs is used to demonstrate the performance of the proposed technique. This system is formulated as follows [Wolovich and Flab, 1969]:

$$G(s) = \begin{bmatrix} \frac{3(s+3)(s+5)}{(s+1)(s+2)(s+4)} & \frac{6(s+1)}{(s+2)(s+4)} \\ \frac{2}{(s+3)(s+5)} & \frac{1}{s+1} \\ \frac{2(s^2+7s+18)}{(s+1)(s+3)(s+5)} & \frac{-2s}{(s+1)(s+3)} \\ \frac{2s+7}{(s+3)(s+4)} & \frac{2s+5}{(s+2)(s+3)} \\ \frac{2(s+5)}{(s+1)(s+2)(s+3)} & \frac{8(s+2)}{(s+1)(s+3)(s+5)} \\ \frac{1}{s+3} & \frac{2(5s^2+27s+34)}{(s+1)(s+3)(s+5)} \end{bmatrix} \tag{30}$$

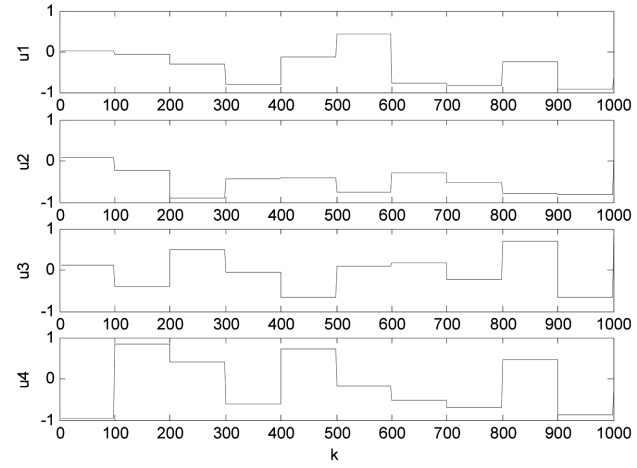
It is a typical MIMO process with interaction. First, the aim is to build up the DynPLS model based on the data. The identification data set contains 1,000 samples obtained at 0.25 sampling time units. Based on the stepwise model-building procedure, the best final predicted dynamic model is depicted as:

$$\begin{bmatrix} y_1^h(k) \\ y_2^h(k) \\ y_3^h(k) \end{bmatrix} = \mathrm{PLS}\left(\begin{bmatrix} x_1(k) \\ x_2(k) \\ x_3(k) \\ x_4(k) \end{bmatrix} \right)$$

$$\begin{bmatrix} y_1(k+1) \\ y_2(k+1) \\ y_3(k+1) \end{bmatrix} = \begin{bmatrix} 0.41y_1(k) - 0.01y_1(k-1) \\ 0.61y_2(k) - 0.22y_2(k-1) \\ 0.23y_3(k) + 0.04y_3(k-1) \end{bmatrix} + \begin{bmatrix} y_1^h(k) \\ y_2^h(k) \\ y_3^h(k) \end{bmatrix} \tag{31}$$



(a)



(b)

Fig. 3. Validation results of the DynPLS model in Example 1: (a) y_1, y_2 and y_3 (b) u_1, u_2, u_3 and u_4.

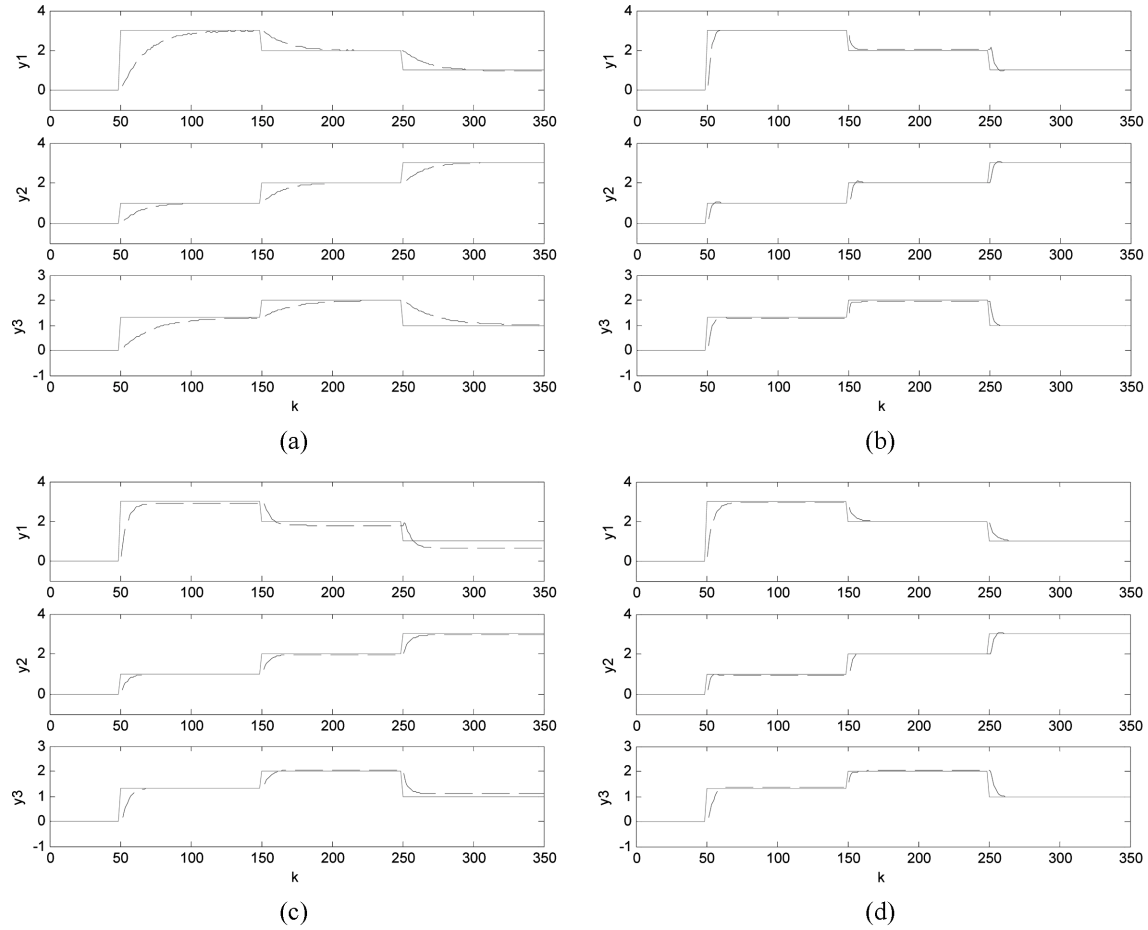


Fig. 4. Control performance of the setpoint change in Example 1 with different inputs: (a) all inputs, (b) $u_1(k)$, $u_2(k)$ and $u_3(k)$, (c) $u_1(k)$, $u_2(k)$ and $u_4(k)$, (d) $u_2(k)$, $u_3(k)$ and $u_4(k)$.

Another 1,000 sets of data which does not come from the training sets are produced in a similar way for validation. The validation results in Fig. 3 show this model closely follows the actual process behavior.

With the built DynPLS model, the setpoint changes can be traced by the on-line updated algorithm that is in control of the process. Two cases with all four inputs and only three inputs are compared. With the proposed control design strategy, this indicates that the proposed multiloop updated algorithm is able to trace the setpoint signal in the MIMO process. Fig. 4 shows the response of the closed-loop system to the different setpoint changes in the reference signal. It is observed that the ouputs in all cases can meet their steady-state values except when the three inputs (u_1, u_2 and u_4) are selected. Fig. 5 shows the updated PID control parameters at each sampling point derived from the system model when all inputs are selected. Table 1 lists the control cost ($\sum_k \sum_i u_i^2(k)$) and the sum square of the error (SSE) of the controlled variables deviated from the set points for different inputs. The costs of the control design and SSE with all four inputs for the nonsquare system are significantly less than those with another three inputs for the square system.

The percentage of variance captured by each PLS component is listed in Table 2. It is observed that three principal components capture over 90% of the variance in the relationships of the MIMO process, which suggests that the process variables are fairly well correlated between inputs and outputs. Here different numbers of control loops based on the number of components are selected to show the control performance (Fig. 6). Since the first component accounts for almost 37% of all the total input variations and 80% of all the total output variations, the control loop based on the first component constitutes the minimum control performance that still barely meets our expectation. Fig. 6(a) shows that the offset occurs due to the model error of the PLS model with only one component even if the controller with an integral mode is used and the response of the controlled score variable is close to the desired setpoint score variable (Fig. 7). With adding the second control loop with the second component, the control performance has been improved a little, but the offset of the output y_2 still exists (Fig. 6(b)). When three control loops based on the first three components are selected, the corresponding control performance is further improved (Fig. 6(c)). However, the improvement is not very significant when the fourth component is added, because the first three components already account for 92% of all the total input variations and 98% of all the total output variations. Therefore, fewer control loops based on the contributions of only a few components in the subspace can be used without a substantial loss of the control performance. The decision depends on how much information (of unaccounted variance) can be removed. Several suggested rules for selecting the number of com-

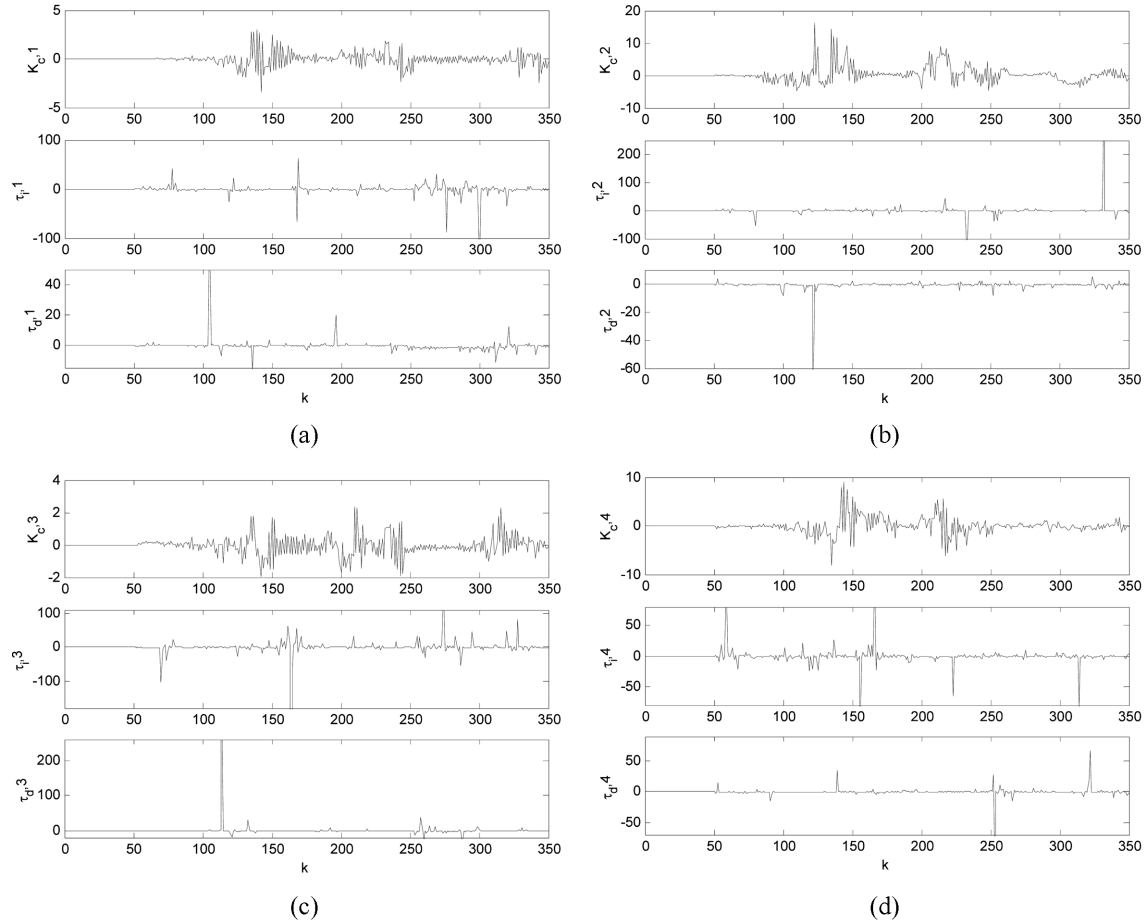


Fig. 5. The multiloop PID controller parameters of the setpoint changes in Example 1: when all inputs are used: (a) loop 1; (b) loop 2; (b) loop 3; (b) loop 4.

Table 1. Total cost of different inputs in Example 1

Control inputs	Input cost	SSE
All four inputs	424.16	27.41
u_1, u_2 and u_3	1756.90	39.05
u_1, u_2 and u_4	6902.10	72.23
u_2, u_3 and u_4	2176.00	47.72

ponents were discussed in the literature [Zwick and Velicer, 1986; Eastment and Krzanowski, 1982].

2. Example 2: Nonlinear pH Neutralization System

A pH neutralization process [Lakshminaraynan et al., 1997; Nahas et al., 1992], which has three input streams and one outlet stream, is considered. The input streams include acid (HNO_3), buffer ($NaHCO_3$) and base (NaOH) streams. The process model consists of two reaction invariants, three nonlinear ordinary equations and one nonlinear algebraic equation, and the input streams are subject to constraints.

Charge balance

$$W_a \equiv [H^+] - [OH^-] - [HCO_3^-] - 2[CO_3^{=}] \quad (32)$$

Carbonate ion balance

$$W_b \equiv [H_2CO_3] + [HCO_3^-] + [CO_3^{=}] \quad (33)$$

Table 2. Percentage of variance captured by each PLS component in Example 1

Component	Percent variance captured by each PLS component			
	Xblock	Total	Yblock	Total
1	37.04	37.04	79.76	79.76
2	34.63	71.68	14.29	94.07
3	20.11	91.78	3.95	98.01
4	8.23	100.00	1.87	99.87

$$\frac{dh}{dt} = \frac{1}{A}(q_1 + q_2 + q_3 - C_v h^{0.5}) \quad (34)$$

$$\frac{dW_{a4}}{dt} = \frac{1}{Ah}[(W_{a1} - W_{a4})q_1 + (W_{a2} - W_{a4})q_2 + (W_{a3} - W_{a4})q_3] \quad (35)$$

$$\frac{dW_{b4}}{dt} = \frac{1}{Ah}[(W_{b1} - W_{b4})q_1 + (W_{b2} - W_{b4})q_2 + (W_{b3} - W_{b4})q_3] \quad (36)$$

$$W_{a4} + 10^{14-pH} + W_{b4}\frac{1 + 2\times 10^{pH-pK_2}}{1 + 10^{pK_1-pH} + 10^{pH-pK_2}} - 10^{-pH} = 0 \quad (37)$$

subject to

$$0 \le q_1(k) \le 30,\ 0 \le q_2(k) \le 30 \quad (38)$$

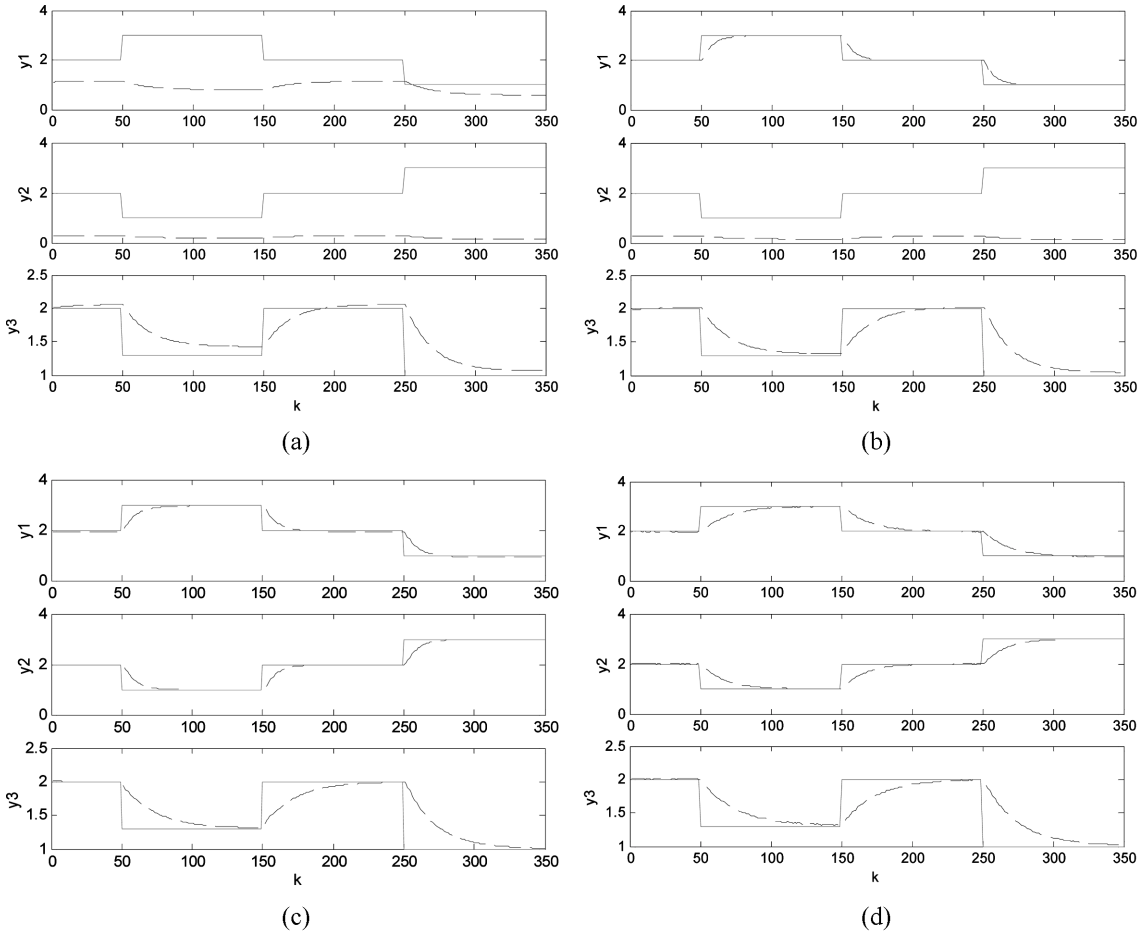


Fig. 6. Control performance of the setpoint change in Example 1 with different number of components: (a) the first one component, (b) the fist two components, (c) the first three components and (d) all components.

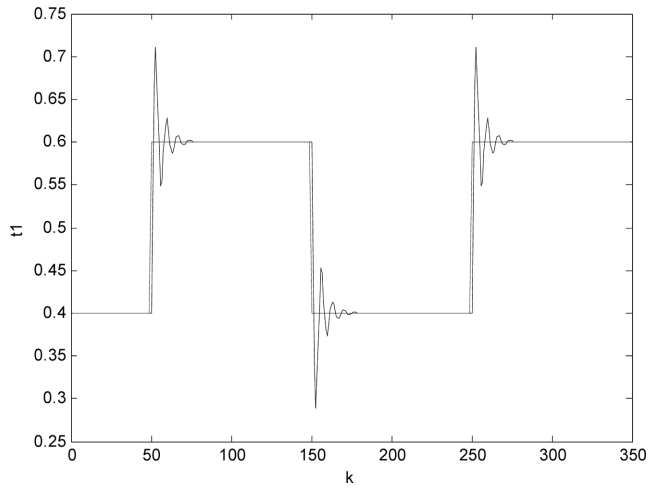


Fig. 7. Projection of the controlled variables in Fig. 6(a) onto the latent space.

In the above equations, h is the liquid level, W_{a4} and W_{b4} are the reaction invariants of the effluent stream, and q_1, q_2 and q_3 are the acid, buffer and base flow rates, respectively. The definition of the other parameters and the nominal operation conditions are listed in Table 3. The objective is to control the pH value and level h in the tank by manipulating the base (q_3) and acid flow rate (q_1).

Table 3. Simulation parameters in Example 2

A=207 cm^2	$W_{b3}=5\times10^{-5}$ M
C_v=8.75 ml cm^{-1} s^{-1}	q_1=16.6 ml s^{-1}
pK1=6.35	q_2=0.55 ml s^{-1}
pK2=10.25	q_3=15.6 ml min^{-1}
$W_{a1}=3\times10^{-3}$ M	[Acid]=0.003 M HNO_3
$W_{a2}=-3\times10^{-2}$ M	[Buffer]=0.03 M $NaHCO_3$
$W_{a3}=-3.05\times10^{-3}$ M	[Base]=0.003 M NaOH

As we know, any linear model has a limited range of validity for the nonlinear process. One way to solve this problem is to use the union of the different local linear models to approximate the desired process. In this study, based on the two desired operating regions (around (i) pH=7.0, h=14.0 and (ii) pH=8.5 , h=12.0), the decomsposition of the modeling problem into two DynPLS models is employed here. Fig. 8 shows the two different local areas and the shaded area covering all the possible steady-state region under the different inputs q_1 and q_3. Before the proposed control statrategy is implemented, two DynPLS models from two open-loop simulation data of the pH system need to be established for these desired operating regions. In this case, the training data set is generated from pseudo-random variation of inputs q_1 and q_3. The duration of each

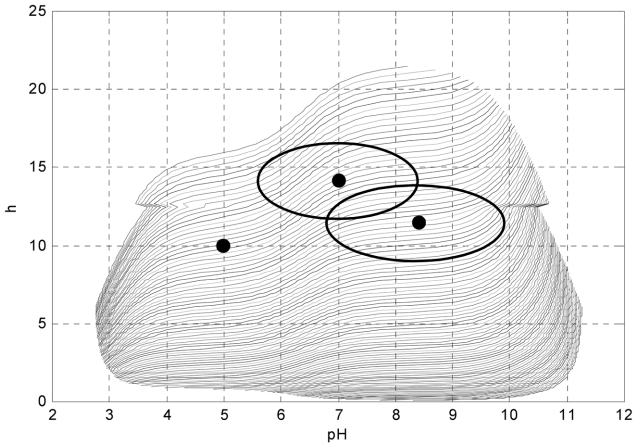


Fig. 8. All the possible steady-state conditions of the pH neutralization system represented by the shaded region. Two ellipses indicated by the dashed lines are the local regions with three standard deviations of h and pH.

variation interval is set to be 20 minutes. The corresponding changes of pH and h are around these two desired operating regions. The two DynPLS models based on the stepwise model-building algorithm are:

Model I: (around pH=7.0, h=14)

$$\hat{y}_1(k+1) = 0.925y_1(k) + y_1^h(k-2)$$
$$\hat{y}_2(k+1) = 1.7032y_2(k) - 0.7236y_2(k-1) - 0.0080y_2(k-2) + y_2^h(k-4) \quad (39)$$

Model II: (around pH=8.5, h=12)

$$\hat{y}_1(k+1) = 1.1066y_1(k) - 0.2818y_1(k-1) + 0.0121y_1(k-2) - 0.0111y_1(k-3) + y_1^h(k)$$
$$\hat{y}_2(k+1) = 1.6773y_2(k) - 0.6817y_2(k-1) - 0.0117y_2(k-1) + y_2^h(k-3) \quad (40)$$

The final optimal combination of these two local models is defined by

$$\mathbf{y}(k) = \sum_{i=1}^{2} w^i \mathbf{y}^i(k) \quad (41)$$

where w^i is the relative validity of each local model and $w_i = \frac{\rho^i}{\sum \rho i^i}$.

ρ^i is the validity function,

$$\rho^i = \exp\left(-\frac{1}{2}\left(\frac{pH - pH^i}{\sigma_{pH}^i}\right)\right) \cdot \exp\left(-\frac{1}{2}\left(\frac{h - h^i}{\sigma_h^i}\right)^2\right) \quad (42)$$

with pH^1=7.0, h^1=14.0, pH^2=8.5, h^2=12.0, $\sigma_{pH}^1=\sigma_{pH}^2$=1.0 and $\sigma_h^1=\sigma_h^2$=1.0. Another data set generated by a similar method is used to verify the prediction capability of the DynPLS models. Fig. 9 shows the approximation capability of the combination models for these testing data. The predicted result exactly follows the acutal process behavior.

In the first testing condition, the control strategy shows the setpoint changes in both the level and the pH value. In the first time period, the values of h and pH are kept at 14 and 7, respectively. At time 100 the setpoints are shifted to the pH value of 8.5, and h value of 12; at time 300 the setpoints are shifted to the pH value of

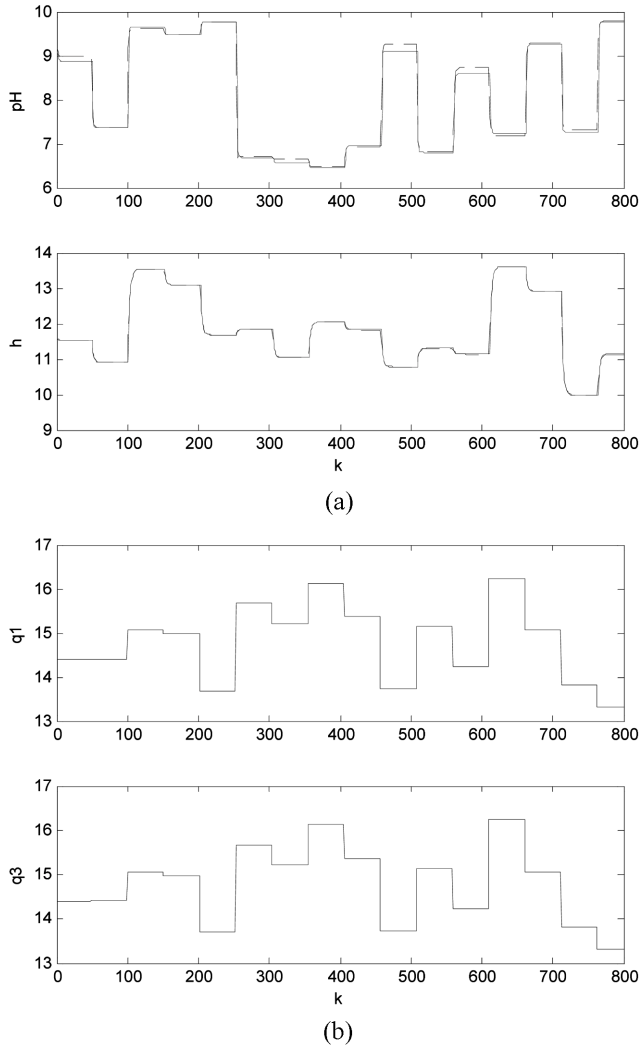


Fig. 9. Validation results of the combinational DynPLS model in Example 2.

6.5, and h value of 16; Fig. 10 demonstrates the setpoint tracking ability of the proposed on-line updated algorithm. The control performance of the combinational DynPLS models is satisfied. QDMC and a nonlinear neural network model predictive control (NNMPC) are also tested for making a fair comparison. QDMC does not have good results because the test region is not covered by the trained model (around pH=8.5 and h=12.0). The manipulated inputs and the corresponding controlled outputs have fairly large variations even if the appropriate value of the Lagrange multiplier constant is included in QDMC in order to have an invertible dynamic matrix and reduce the larger variation of inputs. On the other hand, the decomposition strategy of PLS can remove the components with fewer contributions. The inputs would not have much larger fluctuation even if at time 500 the setpoints are shifted to the new area around the pH value of 5, and h value of 10, which are not covered in the operation regions of the trained model. Furthermore, the controller design of each loop can be directly computed without the invertible problem. When the control performances of NNMPC and DynPLS are compared, although NNMPC is a little better than DynPLS, the former is based on nonlinear optimization and the latter is only used

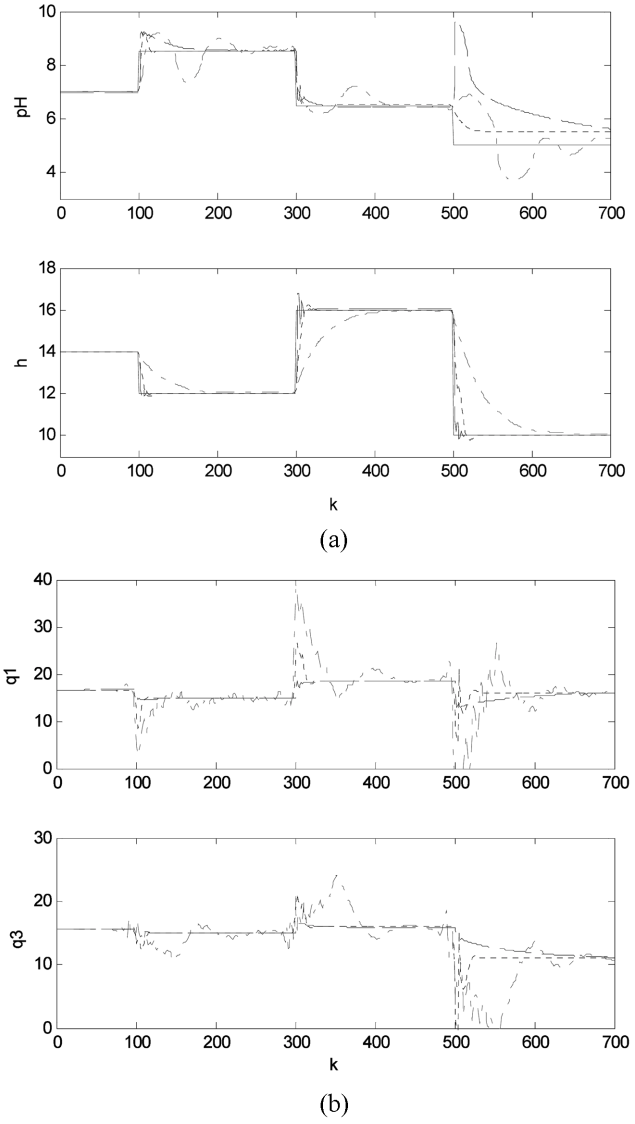


Fig. 10. Control performance of the setpoint changes (solid line) based on QDMC (dashdot line), NNMPC (dotted line) and proposed model (dashed line) in Example 2: (a) and h; (b) q_1 and q_3.

to solve the quadratic objection (Eq. (24)) for each control loop (Eq. (27)). When the setpoints are shifted to the new area after the time point 500, as in the previous discussion, the responses of NNMPC and DynPLS with the modeling error result in the performance deterioration.

Actually, a process with noise always exists. Here the measured h with noise $N(0, 0.3^2)$ and pH with $N(0, 0.2^2)$ are tested. Although the control outputs can follow the setpoints in this situation, they are around the desired setpoints (Fig. 11). The corresponding variations of the controlled variables exist due to the noise measurements to a large extent. Here the control design based on QDMC and NNMPC are also included. The performances of the comparisons of these control designs for the process with noise are the same as those for the process without noise.

In addition to setpoint tracking, the buffer flow rate disturbance is also an important control object in the pH neutralization system.

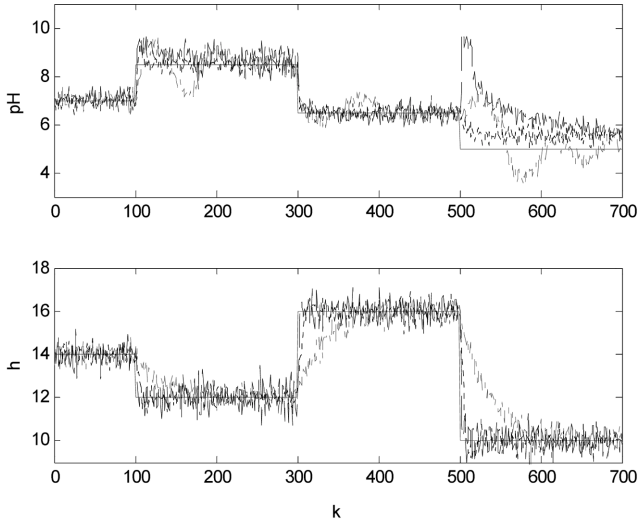


Fig. 11. Control performance of the setpoint changes (solid line) based on QDMC (dashdot line), NNMPC (dotted line) and proposed model (dashed line) in Example 2 when the measurements have noises.

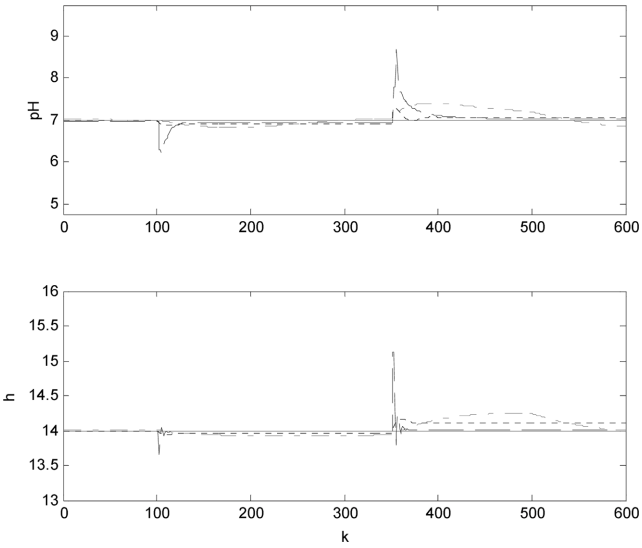


Fig. 12. Control performance of disturbance changes based on QDMC (dashdot line), NNMPC (dotted line) and proposed model (dashed line) in Example 2.

Fig. 12 shows the rejection of buffer disturbance ranges from 0.6 ml/s to 0.2 ml/s at time point 100 and from 0.2 ml/s to 1.5 ml/s at time point 350. Because of the proposed control structure with the integration model, the control performance shows that the proposed DynPLS has a good ability in buffer disturbance rejection without any large offset when compared with the other two methods.

CONCLUSION

In this paper an SISO PID controller design strategy is developed for the design of the MIMO controller system as a substitute for the traditional decoupling design. The proposed method explores many aspects of the control design of the MIMO system, such as

the conceptual decomposition framework in the reduced subspace, the MIMO model development, the sequential training procedures, the optimal control design and applications. This design procedure may lead to a wider range of applications for the multiloop controller structure. The proposed algorithm has the following advantages: (i) It is simple to identify DynPLS since it is not necessary to identify the MIMO system by a sequence of relay identification. (ii) The coupling effect in the MIMO system can be overcome effectively. The PLS structure can be decomposed into several pairs of inputs and outputs, so the number of control loops can be selected based on the variation captured by each pair. (iii) Unlike the sequential tuning of the multiple control loop for the iterative design in each control loop, the adaptive tuning PID controller strategy in the SISO system can be implemented directly and simultaneously onto each loop of the multiloop control design in the MIMO system under the decomposition structure of PLS. The potential of the proposed technique for prediction and process control is demonstrated by means of simulation studies. Modeling and control performed on the large-scale problems and the real lab-scale experiments will be included in our next research.

ACKNOWLEDGMENT

This work was partially supported by the National Science Council of Republic of China.

REFERENCES

Banyasz, C. and Keviczky, L., "Design of Adaptive PID Regulators Based on Recursive Estimation of the Process Parameters," *J. of Proc. Cont.*, **3**, 53 (1993).

Bohim Bobal, V. J. and Prokop, R., "Practice Aspects of Self-tuning Controllers," *International Journal of Adaptive Control and Signal Processing*, **13**, 671 (1999).

Cameron, F. and Seborg, D. E., "A Self-Tuning Controller with a PID Structure," *Int. Journal of Control*, **30**, 401 (1983).

Gawthrop, P. J., "Self-Tuning PID Controllers: Algorithm and Implementation," *IEEE Trans. on Automatic Control*, **31**, 201 (1986).

Garcia, C. E. and Morari, M., "Internal Model Control 1. A Unifying Review and Some New Results," *Ind. Eng. Chem. Process Des. Dev.*, **21**, 308 (1982).

Chen, J. and Yen, J.-H., "Three-Way Data Analysis with Time Lagged Window for On-Line Batch Process Monitoring," *Korean J. Chem. Eng.*, **20**, 1000 (2003).

Chen, J. and Yea, Y., "Neural Network-Based Predictive Control for Multivariable Processes," *Chem. Eng. Comm.*, **189**, 865 (2002).

Eastment, H. T. and Krzanowski, W., "Cross-Validatory Choice of the Number of Components from a Principal Component Analysis," *Technometrics*, **24**, 73 (1982).

Han, I.-S., Kim, M., Lee, C.-H., Cha, W., Ham, B.-K., Jeong, J. H., Lee, H., Chung, C.-B. and Han, C., "Application of Partial Least Squares Methods to a Terephthalic Acid Manufacturing Process for Product Quality Control," *Korean J. Chem. Eng.*, **20**, 977 (2003).

Höskuldsson, A., "PLS Regression Methods," *J. of Chemometrics*, **2**, 211 (1988).

Kashiwagi, H. and Li, Y., "Nonparametric Nonlinear Model Predictive Control," *Korean J. Chem. Eng.*, **21**, 329 (2004).

Kaspar, M. H. and Ray, W. H., "Chemometric Methods for Process Monitoring and High Performance Controller Design," *AIChE J.*, **38**, 1539 (1992).

Kaspar, M. H. and Ray, W. H., "Dynamic PLS Modelling for Process Control," *Chem. Eng. Sci.*, **48**, 3447 (1993).

Kourti, T. and MacGregor, J. F., "Multivariate SPC Methods for Process and Product Monitoring," *J. Quality Technology*, **28**, 409 (1996).

Ku, W., Storer, R. H. and Georgakis, C., "Disturbance Detection and Isolation by Dynamic Principal Component Analysis," *Chemom. Intell. Lab. Syst.*, **30**, 179 (1995).

Lakshminaraynan, S., Shan, S. L. and Nandakumar, K., "Modeling and Control of Multivariable Processes: Dynamic PLS Approach," *AIChE J.*, **43**, 2307(1997).

Luyben, W. L., "Simple Method for Tuning SISO Controllers in Multivariable Systems," *Industrial and Engineering Chemistry*, **25**, 654 (1986).

Nahas, E. P., Henson, M. A. and Seborg, D. E., "Nonlinear Internal Model Control Strategy for Neural Network Models," *Computer and Chemical Engineering*, **12**, 1039 (1992).

Ortega, R. and Kelly, R., "PID Self-Tuners: Some Theoretical and Practice Aspects," *IEEE Trans. on Industrial Electronics*, **31**, 312 (1984).

Oh, S. C. and Yeo, Y.-K., "A Study on the Adaptive Predictive Control Method for Multivariable Bilinear Processes," *Korean J. Chem. Eng.*, **12**, 472 (1995).

Palmor, Z. J., Halevi, Y. and Karsney, N., "Automatic Tuning of Decentralized PID Controller for TITO Processes," *Automatica*, **31**, 1001 (1995).

Proudfoot, C. G., Gawthrop, P. J. and Jacobs, O. L. R., "Self-Tuning PI Control of a pH Neutralization Process," *Proc. of IEE, Pt-D*, **130**, 267 (1983).

Qin, S. J. and McAvoy, T. J., *A Data-Based Process Modeling Approach and Its Applications*, in Proceedings of the 3rd IFAC Symposium on Dynamic and Control of Chemical Reactors, Distillation Columns, and Batch Process, Pergamon, Oxford (1992).

Radke, F. and Isermann, R., "A Parameter-Adaptive PID Controller with Stepwise Parameter Optimization," *Automatica*, **23**, 449 (1987).

Reeves, D. E. and Arkun, Y., "Interaction Measures for Nonsquare Decentralized Control Structures," *AIChE J.*, **35**, 603 (1989).

Shiu, S. L. and Huang, S. H., "Sequential Design Method for Multivariable Decoupling and Multiloop PID Controllers," *Ind. Eng. Chem. Res.*, **37**, 107 (1998).

Wittenmark, B., *Self-Tuning PID Controllers Based on Pole Placement*, Lund Inst. of Technical Report, TFRT-7179 (1979).

Wolovich, W. A. and Flab, P. L., "On the Structure of Multivariable Systems," *SIAM J. Control*, **7**, 437 (1969).

Zhuang, M. and Atherton, D. P., "PID Controller Design for a TITO System," *IEE Proc. on Control and Applications*, **141**, 111 (1994).

Zwick, W. R. and Velicer, W. F., "Comparison of Five Rules for Determining the Number of Components to Retain," *Psychol. Bull.*, **99**, 432 (1986).