Korean J. Chem. Eng., **26**(4), 935-945 (2009)
DOI: 10.1007/s11814-009-0157-z

Development of MBPLS based control for serial operation processes

Junghui Chen† **and Ren-Kai Tsai**

R&D Center for Membrane Technology and Department of Chemical Engineering,
Chung-Yuan Christian University, Chung-Li, Taiwan 320, China
(*Received 14 June 2008 • accepted 9 January 2009*)

Abstract−A control scheme based on the multiblock PLS (MBPLS) model for multi-stage processes (or serially connected processes) is developed. MBPLS arranges a large number of variables into meaningful blocks for each stage of the large-scale system. Two control design strategies, course-to-course (CtC) and within-stage (WS) controls, are proposed for the re-optimization design in the whole multistage course. In CtC, MBPLS control and optimization are done by applying feedback from the finished output quality when one course for all stages is done. It utilizes the information from the current course to improve quality of the next one. In WS, the MBPLS-based re-optimization strategy is developed to explore the possible adjustments of the future inputs at the rest of the stages in order to fix up the disturbances just in time and to maintain the product specification when the current course is finished. The proposed technique is successfully applied to two simulated industrial problems, including a photolithography sequences and a reverse osmosis desalination process, and the advantages of the proposed method are demonstrated.

Key words: Iterative Learning Control, Multistage Process, Partial Least Squares

INTRODUCTION

Traditionally, control design is done in individual stages assuming that each stage is independent of the other stages. With ever-increasing pressure to improve the product quality, the operating system becomes more complicated. It is not enough to consider the control design one stage at a time. In biological wastewater treatment, several sequences of stages are required, including loading, anoxic-anaerobic treatment, aerobic treatment, sedimentation and effluent extraction. If each stage is independent of the other stages in operation, highly inefficient operation may occur. Moreover, it may cause serious process degradation, especially when phosphorus removal is involved. In the manufacture of integrated circuits, the created layers are built up one after another in a sequence of processing steps and units. Many of these units require several hours to complete. Often a wafer lot may require four or five days from the start of fabrication to the end. The serial processes are composed of rather simple operating units. The behavior of individual units could be predicted, but not the serial process consisting of many units placed one after another, in such a way that each unit is influenced by the upstream units. The behavior of the process may become rather complex and very different from what could be expected by looking at the individual units [1]. As concatenation processes become tighter, when a unit is not made correctly, or when a faulty condition damages the production in a unit, the degraded production tends to propagate across the rest of the units. Until the final unit where the poor quality product has been made, it is too late to fix the process. If the product specification can be fixed far upstream of the unit where inspection takes place, the problem of the process can be determined. The appropriate procedures can be worked out to remove the disturbances just at the right unit and to maintain the product specification without waiting for finishing this whole serial process. Therefore, in order to operate the serially connected processes safely and profitably, the development of on-line optimization and the control design are strongly needed.

Despite the viability of the multistage control design, relatively little effort has been devoted to improving control of serially connected processes. The control design of the serially connected processes developed in the past used dynamic programming based optimization tools to handle the performance of the current run across stages [2,3]. The rigorous treatment of exceptional events by scenario-based differential algebraic equation models was also proposed [4]. Because the fatal problem of the methods lies in the fact that the above optimization is obtainable based on a reliable and healthy process model, with aging of the process, the control design based on the models with fixed parameters could not compensate for such process drifts or ramp disturbances. As a result, the effectiveness of the originally designed condition in the real application would be degraded. With rich information of the operating data, empirical models were proposed. Zhang and Li [5]. based on the state space model. presented networked model predictive control for serially connected processes [5]. Because the model parameters are fixed, the strategy implementation would be realistic only for controlled systems whose parameters are changed slowly when the serial stage control is applied

Data-driven multivariable statistical techniques, such as partial least squares (PLS) and multiway partial least squares (MPLS), have been widely used because of their abilities to handle high dimensional and coupled data. They attracted considerable interest in profile tracking control for batch processes. Chen and Wang [6] proposed an iterative learning control strategy based on the PLS method for single batch unit only [6]; Flores-Cerrillo and MacGregor [7] using MPLS developed a terminal iterative learning control strategy for batch-to-batch and within-batch control of final product properties [7]. PLS/MPLS takes advantage of the precision improved by

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

adjusting the number of the latent variables. It also does not require the calculation of the inverse of matrices. The batch-to-batch and on-line control strategies have also been designed by incorporating the information of the previous batches. They exploit the repetitive nature of the batches and the batch-to-batch correlation of the disturbances. However, the above problems focused on the single batch unit. In a typical serial multistage process, there are hundreds of variables to be stored for computation. MPLS works well but it can become unwieldy due to the high cost of memory. The formidable computational effort of the full MPLS model limits its applicability to the processes with moderate number of stages, but the full model of the system, in some cases, is more useful than the sub-models of the individual units, because the control design for the final quality assurance of the whole plant can not be done unless the whole model is available. There is, therefore, a strong temptation to drastically reduce the number of variables, but keeping the whole structure. Recently the multiblock PLS (MBPLS) method was proposed. It is an extension of MPLS except that additional information is available for arranging the variables into blocks with conceptual meaning [8]. Local information from the individual blocks and the global information from the whole block can be obtained simultaneously. It could also zoom in much deeply onto the blocks that were divided by the response variables. The calculated block scores could obtain the local information and the global information simultaneously from the data. The MBPLS method has been used in several applications for monitoring and diagnosis [9-11]. A comprehensive analysis of the MBPLS algorithm can be found [12]. Still, MBPLS was rarely used for the control problem.

Like PLS, MBPLS with only a few principal components could capture most of the characteristics of the system pattern in the multistage process behavior. Therefore, our aim is to develop iterative learning control for the multistage process based on MBPLS. The remainder of this paper is organized as follows: The second section defines the design problem of quality control in the multistage process. The MBPLS modeling procedure is briefly reviewed in Section 3. The MBPLS technique for the decomposition of the relationship between the global and the block scores is also discussed. In Section 4 and 5, MBPLS based course-to-course (CtC) and within-stage (WS) controls are derived separately. In the CtC problem, the manipulated inputs can be adjusted from the measured variables and from the desired quality variables. In the WS problem, using measured variables from the finished stages to develop a control strategy allows rapid rejection or reduction of the disturbances. The effectiveness of the proposed method is demonstrated through two simulated processes in Section 6. Finally, concluding remarks are made.

QUALITY CONTROL OF MULTISTAGE PROCESSES

A serial process with H stages is considered and shown in Fig. 1. The model of stage h can be expressed as

$$\mathbf{y}^h=\mathbf{S}^h(\mathbf{y}^{h-1}, \mathbf{x}^h, \mathbf{d}^h) \tag{1}$$

where $\mathbf{y}^h \in \Re^{M^h}$, $\mathbf{x}^h \in \Re^{N^h}$, $\mathbf{d}^h \in \Re^{L^h}$ are the process output, the manipulated input and the unmeasured disturbance vectors. M^h, N^h and L^h are the number of process outputs, manipulated inputs and unmeasured disturbances at stage h, respectively. $\mathbf{S}^h$ gives the description of the model of the stage h. The output $\mathbf{y}^h$ at stage h is the results of the upstream stage $\mathbf{y}^{h-1}$ and the external inputs ($\mathbf{x}^h$ and $\mathbf{d}^h$). In Eq. (1), the process variables undergo significant changes from one stage to another. It is a noninteracting series since the output of each stage propagates from the input streams to output stream unanimously and the downstream outputs do not affect the upstream outputs. Because of the effect of the propagation, the adjusted inputs ($\mathbf{x}^h$, h=1, …, H) from various stages will not only affect the current stage but also propagate the downstream stages. The aim of the serial process is to regulate the manipulated variables ($\mathbf{x}^h$, h=1, …, H) at each stage and keep the desired product quality at each stage on target ($\mathbf{y}^{h,sp}$, h=1, …, H). The objective function can be formulated as

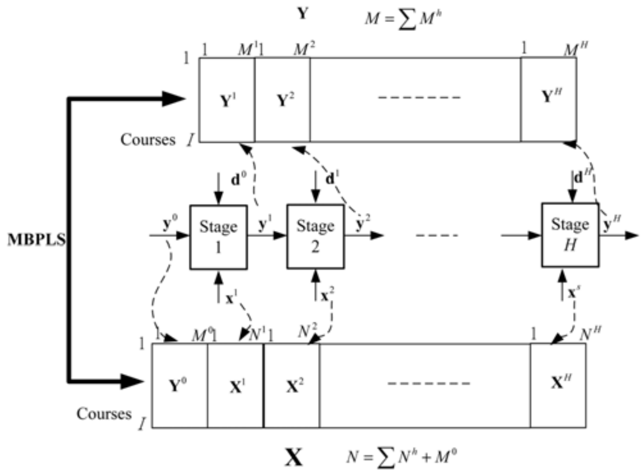

Fig. 1. Typical connection of a multistage process.

$$\min_{x^1,x^2,\ldots,x^H} J = \min_{x^1,x^2,\ldots,x^H} (\mathbf{y}^{sp}-\mathbf{y})(\mathbf{y}^{sp}-\mathbf{y})^T \tag{2}$$

where $\mathbf{y}^{sp}$=[$\mathbf{y}^{1,sp} \cdots \mathbf{y}^{H,sp}$] and $\mathbf{y}$=[$\mathbf{y}^1 \cdots \mathbf{y}^H$] with H stages.

MBPLS MODELING

In MPLS, when the number of variables gets larger, the computation load becomes cumbersome and the model becomes difficult to interpret. It is logical to break the process variables into several blocks based on their similarity or their associated relations in the process; then a projection model is summarized from the contributions of separate blocks. Therefore, MBPLS consists of the sub- and the super-model procedures. In the sub-level of the model, the block models are obtained by multiplying each block score by its block loading. In the super-level of the model, a simple relationship using the block scores as a predictor and a response variable is developed. Basically, MBPLS is more than just MPLS; it can get the local information from the individual blocks and the global information from the whole block simultaneously.

First, the given process data with the manipulated ($\mathbf{X}^h \in \Re^{I\times N^h}$) and the quality ($\mathbf{Y}^h \in \Re^{I\times M^h}$) variables at each stage are collected where I is the number of courses. Fig. 1 shows that the historical data of the multistage process over I courses can be constructed into several two-way arrays. Here assume that there is no time delay from data collection to analysis. The input data matrix contains the blocks based on stage connections:

$$\mathbf{X}=[\mathbf{Y}^0\ \mathbf{X}^1\ \mathbf{X}^2\ \cdots\ \mathbf{X}^H] \tag{3}$$

where $\mathbf{Y}^0 \in \Re^{I\times M^0}$ is a matrix with M^0 off-line collected measurements before entering the whole course. $\mathbf{X} \in \Re^{I\times N}$ and $N=\sum_{h=1}^{H} N^h + M^0$. The output data matrix is the product quality at each stage.

$$\mathbf{Y}=[\mathbf{Y}^1\ \mathbf{Y}^2\ \cdots\ \mathbf{X}^H] \tag{4}$$

where $\mathbf{Y} \in \Re^{I\times M}$ and $M=\sum_{h=1}^{H} M^h$. Usually, each $\mathbf{X}$ and $\mathbf{Y}$ is centered and scaled to unit variance (autoscaling) for each variable at each stage. This essentially puts all variables on an equal basis in modeling. After the MBPLS algorithm is implemented, the $\mathbf{X}$ and $\mathbf{Y}$ matrices are decomposed into the following outer relations,

$$\begin{aligned}\mathbf{X}&=\mathbf{T}^G\mathbf{P}^T+\mathbf{E}\\ \mathbf{Y}&=\mathbf{U}^G\mathbf{Q}^T+\mathbf{F}\end{aligned} \tag{5}$$

where the score matrices of $\mathbf{T}^G$ and $\mathbf{U}^G$ are

$$\mathbf{T}^G=\begin{bmatrix} t^G_{1,1} \cdots t^G_{1,r} \cdots t^G_{1,R} \\ \vdots \quad \vdots \quad \vdots \\ t^G_{i,1} \cdots t^G_{i,r} \cdots t^G_{i,R} \\ \vdots \quad \vdots \quad \vdots \\ t^G_{I,1} \cdots t^G_{I,r} \cdots t^G_{I,R} \end{bmatrix} = \begin{bmatrix} (\mathbf{t}^G_1)^T \\ \vdots \\ (\mathbf{t}^G_i)^T \\ \vdots \\ (\mathbf{t}^G_I)^T \end{bmatrix} \tag{6}$$

and

$$\mathbf{U}^G=\begin{bmatrix} u^G_{1,1} \cdots u^G_{1,r} \cdots u^G_{1,R} \\ \vdots \quad \vdots \quad \vdots \\ u^G_{i,1} \cdots u^G_{i,r} \cdots u^G_{i,R} \\ \vdots \quad \vdots \quad \vdots \\ u^G_{I,1} \cdots u^G_{I,r} \cdots u^G_{I,R} \end{bmatrix} = \begin{bmatrix} (\mathbf{u}^G_1)^T \\ \vdots \\ (\mathbf{u}^G_i)^T \\ \vdots \\ (\mathbf{u}^G_I)^T \end{bmatrix} \tag{7}$$

$\mathbf{P}$ and $\mathbf{Q}$ are loading matrices which show the influences of $\mathbf{X}$ and $\mathbf{Y}$, respectively,

$$\mathbf{P}=[\mathbf{p}_1\ \mathbf{p}_2\ \cdots\ \mathbf{p}_R] \tag{8}$$

and

$$\mathbf{Q}=[\mathbf{q}_1\ \mathbf{q}_2\ \cdots\ \mathbf{q}_R] \tag{9}$$

$\mathbf{E}$ is a residual matrix that is not useful for describing $\mathbf{X}$, and $\mathbf{F}$ contains the part that cannot be related to $\mathbf{Y}$. R is the number of latent variables that can be selected by some heuristic and other statistical methods [13]. The relationship between $\mathbf{X}$ and $\mathbf{Y}$ is represented by a linear algebraic relation between their scores $\mathbf{T}^G$ and $\mathbf{U}^G$. The inner relationship of the score matrices ($\mathbf{U}^G$ and $\mathbf{T}^G$) can be constructed as

$$\mathbf{U}^G=\mathbf{T}^G\mathbf{B} \tag{10}$$

where $\mathbf{B}$ is a diagonal matrix containing the regression coefficients (b_i). Now, after the stage process is calculated by MBPLS, it is broken into a series of univariate regression processes, which is shown in Fig. 2. The transformations of input $\mathbf{x}_i=[\mathbf{y}^0_i\ \mathbf{x}^1_i\ \mathbf{x}^2_i\ \cdots\ \mathbf{x}^H_i]$ and output $\mathbf{y}_i=[\mathbf{y}^1_i\ \mathbf{x}^2_i\ \cdots\ \mathbf{y}^H_i]$ at the course i are expressed respectively,

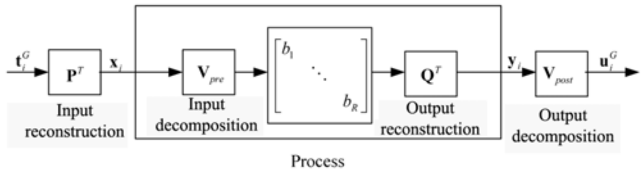

Fig. 2. Diagonalization of the multi-stage process.

$$\mathbf{x}_i\mathbf{V}_{pre}=[\mathbf{t}^G_i]^T \tag{11}$$

$$\mathbf{y}_i\mathbf{V}_{post}=[\mathbf{u}^G_i]^T \tag{12}$$

where $\mathbf{V}_{pre}$ and $\mathbf{V}_{post}$ are pre-multiplied and post-multiplied matrices,

$$\mathbf{V}_{pre}=[\tilde{\mathbf{w}}_1\ \ (\mathbf{I}-\tilde{\mathbf{w}}_1\mathbf{p}_1^T)\tilde{\mathbf{w}}_2 \cdots (\mathbf{I}-\tilde{\mathbf{w}}_1\mathbf{p}_1^T)(\mathbf{I}-\tilde{\mathbf{w}}_2\mathbf{p}_2^T)\ldots(\mathbf{I}-\tilde{\mathbf{w}}_{R-1}\mathbf{p}_{R-1}^T)\tilde{\mathbf{w}}_R] \tag{13}$$

$$\mathbf{V}_{post}=[\mathbf{q}_1\ \ (\mathbf{I}-\mathbf{q}_1\mathbf{q}_1^T)\mathbf{q}_2 \cdots (\mathbf{I}-\mathbf{q}_1\mathbf{q}_1^T)(\mathbf{I}-\mathbf{q}_2\mathbf{q}_2^T)\ldots(\mathbf{I}-\mathbf{q}_{R-1}\mathbf{q}_{R-1}^T)\mathbf{q}_R] \tag{14}$$

where $\tilde{\mathbf{w}}_r$, r=1, 2, ⋯, R is the loading vector for $\mathbf{X}$ [14]. The resulting transformed inputs $\mathbf{t}^G_i$ and $\mathbf{u}^G_i$ outputs are now completely decoupled.

The score ($\mathbf{t}^G_i$) is called global-score. With the global-score, it is possible to get the local-score, which is also called block-score ($\mathbf{t}^h_r$, h=1, ⋯, H), by the global weight ($\mathbf{w}^G_r$)

$$\begin{bmatrix} t^G_{1,r} \\ \vdots \\ t^G_{i,r} \\ \vdots \\ t^G_{I,r} \end{bmatrix} = [\mathbf{t}^1_r\ \cdots\ \mathbf{t}^h_r\ \cdots\ \mathbf{t}^H_r]\mathbf{w}^G_r,\ \ r=1,\ \cdots,\ R \tag{15}$$

where $\mathbf{t}^h_r=[t^h_{1,r}\ t^h_{2,r}\ \cdots\ t^h_{I,r}]^T$, h=1, 2, ⋯, H and $\mathbf{w}^G_r=[w^G_{1,r}\ w^G_{2,r}\ \cdots\ w^G_{H,r}]^T$. Thus, the element of the global-score $\mathbf{t}^G_i$ (, $t^G_{i,r}$) in the above equation can be expressed in terms of the element of each block-score ($t^h_{i,r}$)

$$t^G_{i,r}=\sum_{h=1}^{H} t^h_{i,r} w^G_{h,r} \tag{16}$$

where $t^h_{i,r}$ is computed by the projection of the stage input ($\mathbf{X}^h$) onto the local block weight. Rewriting Eq. (12) to express it in terms of the elements of the matrix equations, it has the following form

$$\begin{bmatrix} u^G_{i,1} \\ \vdots \\ u^G_{i,r} \\ \vdots \\ u^G_{i,R} \end{bmatrix} = [\mathbf{y}^1_i\ \cdots\ \mathbf{y}^h_i\ \cdots\ \mathbf{y}^H_i] \begin{bmatrix} \mathbf{V}^1_{post,1} \cdots \mathbf{V}^1_{post,r} \cdots \mathbf{V}^1_{post,R} \\ \vdots \quad \vdots \quad \vdots \\ \mathbf{V}^h_{post,1} \cdots \mathbf{V}^h_{post,r} \cdots \mathbf{V}^h_{post,R} \\ \vdots \quad \vdots \quad \vdots \\ \mathbf{V}^H_{post,1} \cdots \mathbf{V}^H_{post,r} \cdots \mathbf{V}^H_{post,R} \end{bmatrix} \tag{17}$$

In the same manner, the elements of the global-score $\mathbf{u}^G_i$ (, $u^G_{i,r}$) can be expressed in terms of the elements of each block-score ($u^h_{i,r}$)

$$u^G_{i,r}=\sum_{h=1}^{H} u^h_{i,r} \tag{18}$$

where the block-score ($u^h_{i,r}$, h=1, ⋯, H) is defined as $u^h_{i,r}=\mathbf{y}^h_i\mathbf{V}^h_{post,r}$. The detailed MBPLS algorithm for modeling and prediction has been well discussed in chemeometrics literature [12,15,16]. The MBPLS algorithm can define the relations of the block scores and the global scores. With the relations of the stages in mind, the control algorithm can be easily developed to separately design the manipulated variables at each stage and keep the stage qualities and the final product qualities at the desired value.

MBPLS BASED MULTISTAGE CONTROL DESIGN

The aim of the control design for the multistage process is to seek appropriate manipulated inputs at each stage $\mathbf{x}^h$ that causes the product qualities at each stage $\mathbf{y}^h$ to match the desired target $\mathbf{y}^{h,sp}$. Based on the MBPLS, the multistage system is decomposed into several pairs of the input scores and the output scores. The multistage control design is conducted for each pair separately. The block diagram of the multistage control system is shown in Fig. 3.

1. MBPLS-based Decoupling of MIMO Multistage into SISO One

First, $\mathbf{y}^{sp}=[\mathbf{y}^{1,sp} \ \cdots \ \mathbf{y}^{H,sp}]$ and $\mathbf{y}_i$ are decomposed into the lower dimensional space via the MBPLS deflation procedure, $\mathbf{y}^{sp}=\mathbf{y}^{sp}_{MBPLS}+\mathbf{f}^{sp}$ and $\mathbf{y}_i=\mathbf{y}_{MBPLS,i}+\mathbf{f}_i$. Now the control objective, (Eq. (2)), can be represented by

$$\min_{\mathbf{x}_i^1,\mathbf{x}_i^2,\cdots,\mathbf{x}_i^H} \mathrm{J} = \min_{\mathbf{x}_i^1,\mathbf{x}_i^2,\cdots,\mathbf{x}_i^H} \left\| \mathbf{y}^{sp}_{MBPLS} - \mathbf{y}_{MBPLS,i} \right\|^2 \tag{19}$$

Here assume that the difference of the projection errors ($\mathbf{f}^{sp}$ and $\mathbf{f}_i$) which comes from the random noise or the insignificant disturbance variations can be neglected. Substituting $\mathbf{y}^{sp}_{MBPLS}=\sum_{r=1}^{R} \mathrm{u}_r^{G,sp}\mathbf{q}_r^T$ and $\mathbf{y}_{MBPLS,i}=\sum_{r=1}^{R} \mathrm{u}_{i,r}^{G}\mathbf{q}_r^T$ into Eq. (19),

$$\begin{aligned}
\min_{\mathbf{x}_i^h, h=1,\cdots,H} \mathrm{J} &\cong \min_{\mathbf{x}_i^h, h=1,\cdots,H} \left\| \sum_{r=1}^{R} (\mathrm{u}_r^{G,sp} - \mathrm{u}_{i,r}^{G})\mathbf{q}_r^T \right\|^2 \\
&\le \min_{\mathbf{x}_i^h, h=1,\cdots,H} \sum_{r=1}^{R} \|\mathbf{q}_r\|^2 \sum_{r=1}^{R} (\mathrm{u}_r^{G,sp} - \mathrm{u}_{i,r}^{G})^2 \\
&= \min_{t_{i,r}^G, r=1,\cdots,R} [\mathrm{J}_1 + \mathrm{J}_2 + \cdots \cdots + \mathrm{J}_R]
\end{aligned}$$

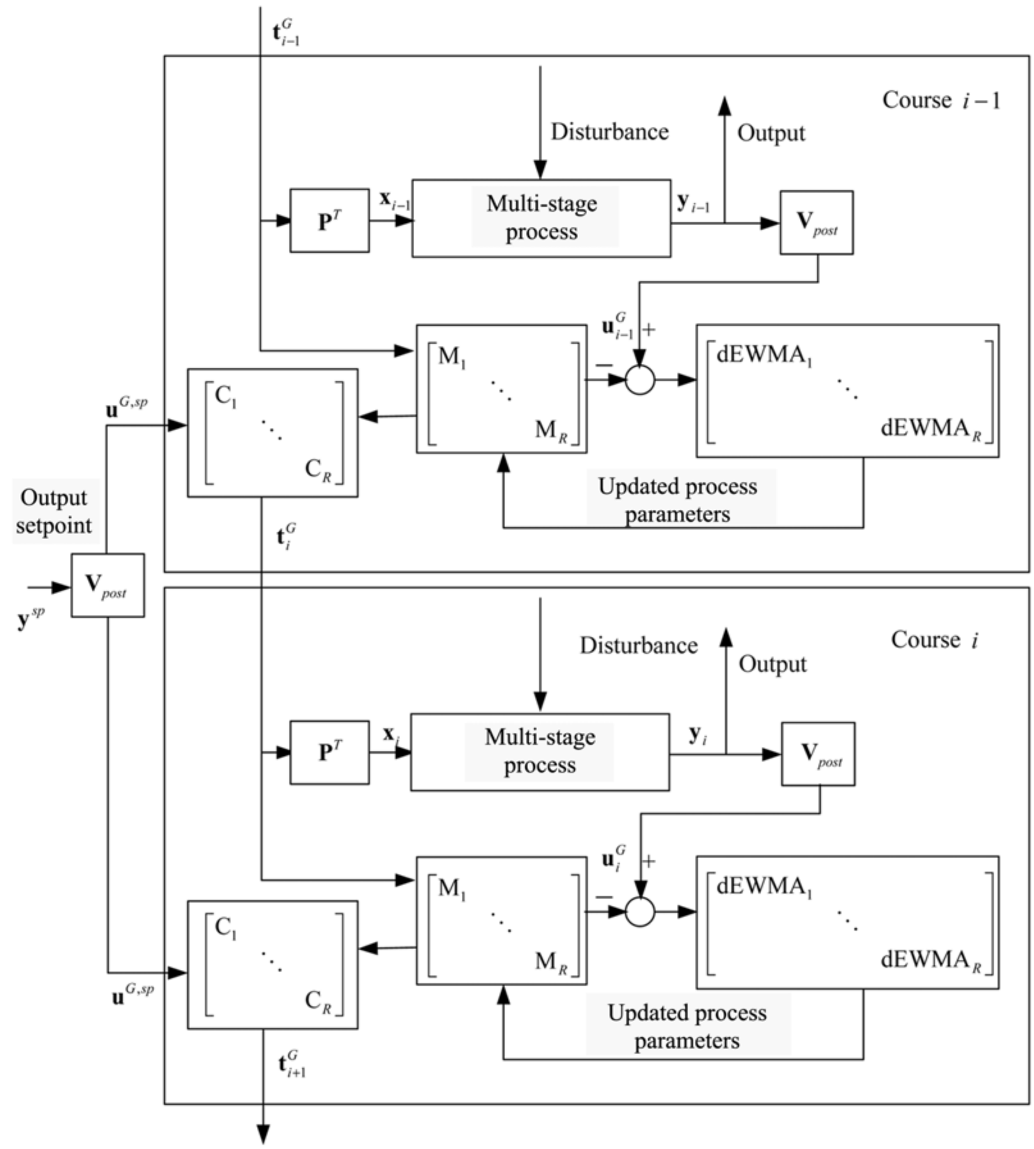

Fig. 3. MBPLS based control structure using the course-to-course strategy.

$$=\left[\min_{t_{i,1}^G} J_1 + \min_{t_{i,2}^G} J_2 + \cdots\cdots + \min_{t_{i,R}^G} J_R\right] \tag{20}$$

where $J_r \equiv \|\mathbf{q}_r\|^2(u_r^{G,sp}-u_{i,r}^G)^2$ and the design variables ($\mathbf{x}^h$, h=1, 2, ⋯, H) are transformed into the score variables ($\mathbf{t}_i^G$) in the subspace via Eq. (11). The objective function is decomposed into R sub-objective functions in the lower dimensional subspace, $J=\sum_{r=1}^{R} J_r$. Unlike the whole system responses lumped together with the risk of the overdetermined or underdetermined problems, R score variables ($t_{i,r}^G$, r=1, 2, ⋯, R) are separately designed because MPLS can pair the inputs and the corresponding outputs.

2. CtC Control Design for Each Component

Each sub-objective (J_r) is rearranged into

$$\min_{t_{i,r}^G} J_r = \|\mathbf{q}_r\|^2 \min_{t_{i,r}^G}(u_r^{G,sp}-u_{i,r}^G)^2 \tag{21}$$

Because of the persistent disturbance, the quality MBPLS models ($u_{i,r}^G = b_r t_{i,r}^G$) developed from Eq. (10) do not always have accurate correlation. When the model is not good enough, the effectiveness of the control design becomes questionable. The conventional dEWMA control design is used here to compensate for the model error. This means that the observer for the output score ($\hat{\mathbf{u}}_i^G$) with the following form is assumed,

$$\hat{\mathbf{u}}_i^G = \begin{bmatrix} \hat{u}_{i,1}^G \\ \vdots \\ \hat{u}_{i,R}^G \end{bmatrix} = \begin{bmatrix} a_{i,1} \\ \vdots \\ a_{i,R} \end{bmatrix} + \begin{bmatrix} b_1 & 0 & 0 \\ 0 & \ddots & 0 \\ 0 & 0 & b_R \end{bmatrix} \begin{bmatrix} t_{i,1}^G \\ \vdots \\ t_{i,R}^G \end{bmatrix} + \begin{bmatrix} d_{i,1} \\ \vdots \\ d_{i,R} \end{bmatrix} \tag{22}$$
$$=\mathbf{a}_i+\mathbf{B}\mathbf{t}_i^G+\mathbf{d}_i$$

This equation represents the model $\mathbf{M}=\{M_r\}_{r=1,\cdots,R}$ shown in Fig. 3. The bias term $\mathbf{a}_i$ and the trend estimation filter $\mathbf{d}_i$ are recursively updated by

$$\mathbf{a}_i=\lambda_1(\mathbf{u}_{i-1}^G-\mathbf{B}\mathbf{t}_{i-1}^G)+(\mathbf{I}-\lambda_1)\mathbf{a}_{i-1} \tag{23}$$

and

$$\mathbf{d}_i=\lambda_2(\mathbf{u}_{i-1}^G-\mathbf{B}\mathbf{t}_{i-1}^G-\mathbf{a}_{i-1})+(\mathbf{I}-\lambda_2)\mathbf{d}_{i-1} \tag{24}$$

where $\lambda_1=diag(\lambda_{1,1} \ldots \lambda_{1,R})$ and $\lambda_2=diag(\lambda_{2,1} \ldots \lambda_{2,R})$ are the weight matrices. The above two equations are for the updated model **dEWMA**=$\{dEWMA_r\}_{r=1,\cdots,R}$ shown in Fig. 3. If the parameters $\mathbf{a}_i$ and $\mathbf{d}_i$ from course i−1 to i do not change and the model (Eq. (22)) is known to be the true system description, the minimum mean square error control strategy that achieves the target value $\mathbf{u}^{G,sp}=[u_1^{G,sp} \cdots u_R^{G,sp}]^T$ can fix the input score at the value

$$\mathbf{t}_i^G = \begin{bmatrix} t_{i,1}^G \\ \vdots \\ t_{i,R}^G \end{bmatrix} = \mathbf{B}^{-1}(\mathbf{u}^{G,sp}-\mathbf{a}_i-\mathbf{d}_i) \tag{25}$$

This equation represents the controller $\mathbf{C}=\{C_r\}_{r=1,\cdots,R}$ shown in Fig. 3. Thus, the input variables for the i course run are

$$\hat{\mathbf{x}}_i=(\mathbf{t}_i^G)^T\mathbf{P}^T \tag{26}$$

The updated input variables $\hat{\mathbf{x}}_i=[\hat{\mathbf{y}}_i^0 \ \hat{\mathbf{x}}_i^1 \ \cdots \ \hat{\mathbf{x}}_i^H]$, including $\hat{\mathbf{y}}_i^0$ which represents the designed conditions to feed stage 1, can be implemented on the next run of the whole course.

[Note] Suppose that the whole course operation with a linear drift of the latent variables in the reduced space is represented by

$$\mathbf{u}_i^G=\mathbf{a}^p+\mathbf{B}\mathbf{t}_i^G+\delta^p i \tag{27}$$

where the score output is systematically drifting away by a size of $\delta^p=[\delta_1^p \cdots \delta_R^p]^T$ per course run. To see how the course asymptotic behavior converges when the above control design is implemented, let $\mathbf{e}_i \equiv \mathbf{y}_{MPLS}^{sp}-\mathbf{y}_{MPLS,i}=\mathbf{e}_i^u\mathbf{Q}^T$ and $\mathbf{e}_i^u=(\mathbf{u}^{sp}-\mathbf{u}_i)^T$. If the proposed control structure is applied, the tracking error in the subspace derived from Eqs. (25) and (27) yields

$$\mathbf{e}_{i+1}^u=\mathbf{e}_i^u(\mathbf{I}-\lambda_1-\lambda_2)+(\lambda_1\mathbf{d}_i-\delta^p) \tag{28}$$

where $\mathbf{d}_i$ can be represented by

$$\mathbf{d}_i=\lambda_2((\mathbf{I}-\lambda_1)^i(\mathbf{a}^p+\delta^p-\mathbf{a}_0)+(\mathbf{I}+(\mathbf{I}-\lambda_1)+\cdots(\mathbf{I}-\lambda_1)^{i-1})\delta^p)+(\mathbf{I}-\lambda_2)\mathbf{d}_{i-1} \tag{29}$$

As the course approaches infinity,

$$E(\mathbf{d}_i)=\lambda_2\lim_{i\to\infty}(\mathbf{S}\delta^p)+(\mathbf{I}-\lambda_2)E(\mathbf{d}_{i-1})=\lambda_1^{-1}\delta^p \tag{30}$$

where $\mathbf{S}=diag\left(\frac{1}{\lambda_{1,1}} \ \cdots \ \frac{1}{\lambda_{1,R}}\right)$. Substituting Eq. (30) into Eq. (28),

$$\mathbf{e}_{i+1}^u=\mathbf{e}_i^u(\mathbf{I}-\lambda_1-\lambda_2) \tag{31}$$

The asymptotic behavior of the output error can be obtained

$$E(\mathbf{e}_i)=E(\mathbf{e}_i^u\mathbf{Q}^T)=0 \tag{32}$$

It can be seen that the asymptotic behavior using the proposed control scheme can definitely converge to the desired targets.

MBPLS BASED WITHIN-STAGE CONTROL DESIGN

MBPLS based iterative control uses a feedforward controller to design the operating manipulated input variables for the whole course run. It can only improve the performance of the next course run. Because the stage process is conducted sequentially, the corrective control actions for the remaining course of the stage process operation should be taken when an unknown disturbance presents at some stages and its effects could be sensed from the stage measurements. They will minimize the bad effects of the unknown disturbance on the product qualities for the rest of the stages.

When the system is currently run at the particular stage h shown in Fig. 4, the measured variables $\mathbf{x}_i=[\mathbf{x}_i^p \ \mathbf{x}_i^f]$ are partly available. The first part ($\mathbf{x}_i^p=[\mathbf{y}_i^0 \ \mathbf{x}_i^1 \ \mathbf{x}_i^2 \ \cdots \ \mathbf{x}_i^h]$) is the data already measured until the current stage h. The other part, ($\mathbf{x}_i^f=[\mathbf{x}_i^{h+1} \ \mathbf{x}_i^{h+2} \ \cdots \ \mathbf{x}_i^H]$), is the future control actions from the next stage to the end of the course run. Now, the computed adjustments from Eq. (26) should not be directly implemented on the rest of the stages if the measured variables, $\mathbf{y}_i^p=[\mathbf{y}_i^1 \ \mathbf{y}_i^2 \ \cdots \ \mathbf{y}_i^h]$, that have already occurred are not equal to $\mathbf{y}^{p,sp}=[\mathbf{y}^{1,sp} \ \mathbf{y}^{2,sp} \ \cdots \ \mathbf{y}^{h,sp}]$ because the unknown upset is induced. If the abnormal condition cannot be fixed just in time, the off-specification product would exist all the way to the final stage. To correct the off-specification far upstream of the stage before inspecting the finished product, the process model should be updated and the control design should be refined at the right time. To correct the mismatch of the already measured ($\mathbf{y}_i^p$) and the desired ($\mathbf{y}^{sp}$) variables, the bias ($\mathbf{a}_i$) and the trend ($\mathbf{d}_i$) terms should be first updated based on the already measured variables ($\mathbf{y}^p$). The updated dEWMA equa-

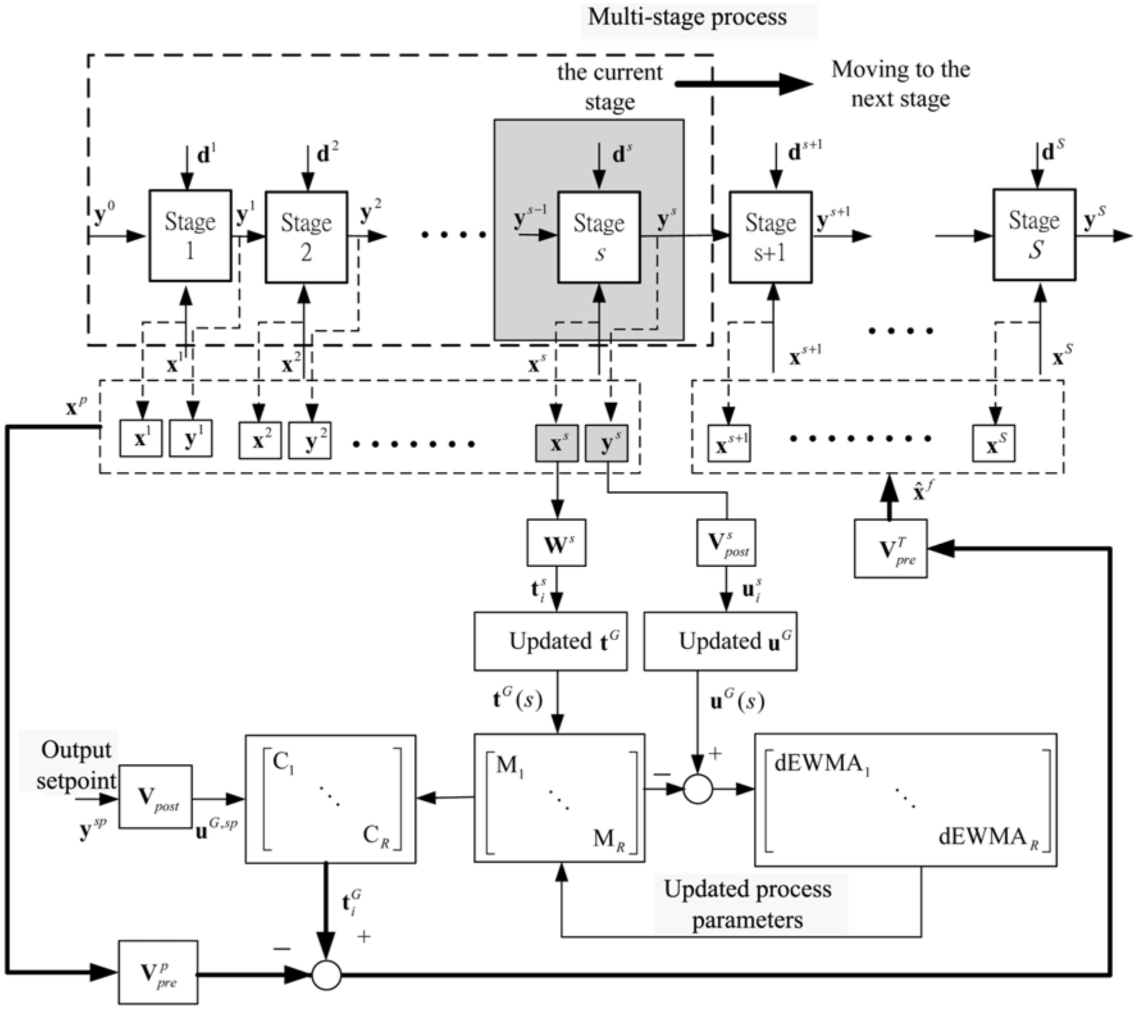

Fig. 4. MPLS based control structure stage by stage.

tion based on the prediction error and the parameter at stage h is given by,

$$\mathbf{a}_i(\mathrm{h}+1)=\lambda_3(\mathbf{u}_i^G(\mathrm{h})-\mathbf{B}\mathbf{t}_i^G(\mathrm{h})-\mathbf{d}_i)+(\mathbf{I}-\lambda_3)\mathbf{a}_i(\mathrm{h}) \tag{33}$$

where the trend parameter term ($\mathbf{d}$) has no change at the current course and it is updated only from course to course. From Eqs. (16) and (18), $\mathbf{u}_i^G(\mathrm{h})$ and $\mathbf{t}_i^G(\mathrm{h})$ for the whole stage can be separately decomposed into the combination of $\mathbf{u}_i^h$ and $\mathbf{t}_i^h$ of the individual stage, h=1, 2, …, H,

$$\begin{aligned}\mathbf{u}_i^G(\mathrm{h})&=\underbrace{\sum_{s=1}^{h-1}\mathbf{u}_i^s}_{\text{past stags}}+\underbrace{\mathbf{u}_i^h}_{\substack{\text{current}\\\text{stage}}}+\underbrace{\sum_{s=h+1}^{H}\mathbf{u}_i^s}_{\text{future stages}}\\ \mathbf{t}_i^G(\mathrm{h})&=\underbrace{\sum_{s=1}^{h-1}\mathbf{t}_i^s}_{\text{past stags}}+\underbrace{\mathbf{t}_i^h}_{\substack{\text{current}\\\text{stage}}}+\underbrace{\sum_{s=h+1}^{H}\mathbf{t}_i^s}_{\text{future stages}}\end{aligned} \tag{34}$$

with $\mathbf{u}_i^h=[\mathrm{u}_{i,1}^h\ \mathrm{u}_{i,2}^h\ \cdots\ \mathrm{u}_{i,R}^h]^T$ and $\mathbf{t}_i^h=[\mathrm{t}_{i,1}^h\mathrm{w}_{s,1}^G\ \mathrm{t}_{i,2}^h\mathrm{w}_{s,2}^G\ \cdots\ \mathrm{t}_{i,R}^h\mathrm{w}_{s,R}^G]^T$. Since $\mathbf{t}_i^G(\mathrm{h})$ affects $\mathbf{u}_i^s$ (s=h+1, ⋯, H) in the future stages, $\mathbf{t}_i^G(\mathrm{h})$ is not available at stage h. The unknown $\mathbf{u}_i^s$ is assumed to be in accordance with $\mathbf{u}_{i-1}^s$ as computed from the previous course. Therefore, with the measured variables available at stage h, $\mathbf{u}_i^h$ and $\mathbf{t}_i^h$, the model ($\mathbf{a}_i$) can be updated (Eq. (33)) without going through all the stages. Now the desired input score is re-computed (Eq. (25)). This will appropriately re-adjust the manipulated variable, which is explained in the next paragraph, to reach the desired measured variables for the rest of the stages along the stage axis.

When the process is conducted at stage (h), the manipulated variables ($\mathbf{x}_i^f$) for the rest of the stages ($\mathbf{x}_i^{h+1}$, ⋯, $\mathbf{x}_i^H$) can be computed from Eq. (26), but they cannot be directly implemented because the computed $\hat{\mathbf{x}}_i^p$ may not be equal to the measured variables $\mathbf{x}_i^p$, which already occurred. The error (δ^h) of $\hat{\mathbf{x}}_i^p$ and $\mathbf{x}_i^p$ is defined as

$$\delta^h\equiv\hat{\mathbf{x}}_i^p-\mathbf{x}_i^p \tag{35}$$

To achieve the desired target qualities, the computed score value $\mathbf{t}_i^G=[\mathrm{t}_{i,1}^G\ \cdots\ \mathrm{t}_{i,R}^G]^T$ should still be applied. Substituting Eq. (35) in Eq. (26),

$$(\mathbf{t}_i^G)^T=\hat{\mathbf{x}}_i\mathbf{V}_{pre}=[\hat{\mathbf{x}}_i^p\ \ \hat{\mathbf{x}}_i^f]\begin{bmatrix}\mathbf{V}_{pre}^p\\ \mathbf{V}_{pre}^f\end{bmatrix}=\mathbf{x}_i^p\mathbf{V}_{pre}^p+\delta^h\mathbf{V}_{pre}^p+(\mathbf{t}_i^G)^T(\mathbf{P}^f)^T\mathbf{V}_{pre}^f \tag{36}$$

where $\mathbf{P}=\begin{bmatrix}\mathbf{P}^p\\ \mathbf{P}^f\end{bmatrix}$ is the loading matrix for $\mathbf{X}$, and

$$\hat{\mathbf{x}}_i = [\hat{\mathbf{x}}_i^p \;\; \hat{\mathbf{x}}_i^f] = \lfloor (\mathbf{t}_i^G)^T (\mathbf{P}^p)^T \;\; (\mathbf{t}_i^G)^T (\mathbf{P}^f)^T \rfloor \tag{37}$$

Assume the adjusted error term ($\delta^h \mathbf{V}^p_{pre}$) in the $(\mathbf{P}^f)^T \mathbf{V}^f_{pre}$ space is defined as

$$\delta^h \mathbf{V}^p_{pre} \equiv \alpha (\mathbf{P}^f)^T \mathbf{V}^f_{pre} \tag{38}$$

Thus, rewrite Eq. (36) and the input values in the score space are

$$((\mathbf{t}_i^G)^T + \alpha) = ((\mathbf{t}_i^G)^T - \mathbf{x}_i^p \mathbf{V}^p_{pre})((\mathbf{P}^f)^T \mathbf{V}^f_{pre})^{-1} \tag{39}$$

Therefore, in order to keep the computed score value $\mathbf{t}_i^G$, the variables $\hat{\mathbf{x}}_i^f$ should be re-adjusted as

$$\begin{aligned}\hat{\mathbf{x}}_i^f &= [\mathbf{x}_i^{h+1} \;\; \mathbf{x}_i^{h+2} \;\; \cdots \;\; \mathbf{x}_i^H] \\ &= ((\mathbf{t}_i^G)^T + \alpha)(\mathbf{P}^f)^T = ((\mathbf{t}_i^G)^T - \mathbf{x}_i^p \mathbf{V}^p_{pre})((\mathbf{P}^f)^T \mathbf{V}^f_{pre})^{-1} (\mathbf{P}^f)^T\end{aligned} \tag{40}$$

Only the manipulated variables $\mathbf{x}_i^{h+1}$ of stage h+1 selected from $\hat{\mathbf{x}}_i^f = [\mathbf{x}_i^{h+1} \;\; \mathbf{x}_i^{h+2} \;\; \cdots \;\; \mathbf{x}_i^H]$ are actually implemented. Then the process outputs of stage h+1 are collected to recursively update the model. The above procedure is repeated for the rest of the stages until the whole course is finished.

[NOTE] In the WS control environment, the output error ($\mathbf{e}^u_{i+1}$(h)) at the adjacent two stages (h and h+1) can be represented by

$$\mathbf{e}^u_{i+1}(\mathrm{h+1}) = \mathbf{e}^u_{i+1}(\mathrm{h}) - (\mathbf{B} \Delta \mathbf{t}^G_{i+1}(\mathrm{h+1}))^T \tag{41}$$

where $\Delta \mathbf{t}^G_{i+1}(\mathrm{h+1}) \equiv \mathbf{t}^G_{i+1}(\mathrm{h+1}) - \mathbf{t}^G_{i+1}(\mathrm{h})$. Based on Eq. (25), the adjustment of $\Delta \mathbf{t}^G_{i+1}(\mathrm{h+1})$ is

$$\Delta \mathbf{t}^G_{i+1}(\mathrm{h+1}) = \mathbf{B}^{-1}(-\mathbf{a}_{i+1}(\mathrm{h+1}) + \mathbf{a}_{i+1}(\mathrm{h})) \tag{42}$$

Substituting Eq. (42) into Eq. (41) yields

$$\mathbf{e}^u_{i+1}(\mathrm{h+1}) = \mathbf{e}^u_{i+1}(\mathrm{h})(\mathbf{I} - \lambda_3) \tag{43}$$

Thus, if the whole process with H stages is run, the measurement error is

$$\mathbf{e}^u_{i+1} = \mathbf{e}^u_{i+1}(\mathrm{H}-1) = \mathbf{e}^u_{i+1}(1)(\mathbf{I} - \lambda_3)^{H-2} \tag{44}$$

where $\mathbf{e}^u_{i+1}(1) = \mathbf{e}^u_i(\mathbf{I} - \lambda_1 - \lambda_2)$. The output error sequence between the adjacent two courses using within-stage control is

$$\mathbf{e}^u_{i+1} = \mathbf{e}^u_i(\mathbf{I} - \lambda_1 - \lambda_2)(\mathbf{I} - \lambda_3)^{H-2} \tag{45}$$

With the comparison of CtC control (Eq. (31)) and WS control (Eq. (45)), it is obvious that the convergence of WS between the adjacent two courses is faster.

ILLUSTRATIVE EXAMPLES

1. Example 1: Photolithography Sequences

A three-stage photolithography sequence used in semiconductor manufacturing provides a good example of the possible applications of the multistage control design. The model was developed by Leang et al. [17,18]. In this photolithography operation, three steps are needed. Because of the proprietary nature of the third step, only the models from the first two stages are considered here.

(i) Spin-coat and bake for the SVG 8626 wafer track

$$\begin{aligned}&\mathrm{M}^1 = 0.91 + 1.61 \times 10^{-3}\mathrm{BTE} - 2.10 \times 10^{-5}\mathrm{SPS} + \varepsilon_{11} \\ &\mathrm{T}_{res} = 1291.98 + 928233(\mathrm{SPS})^{-0.5} - 1.62\mathrm{BTI} - 19.49\mathrm{BTE} + \varepsilon_{12}\end{aligned} \tag{46}$$

where SPS is spin speed (rpm); BTI, baking time (second); BTE, baking temperature (Celsius); T_{res}, resist thickness (Angstroms) and M^1, relative photoactive compound concentration. $\varepsilon_{1,1} \sim \mathrm{N}(0, 0.015^2)$ and $\varepsilon_{1,2} \sim \mathrm{N}(0, 15^2)$ are the measurement noises.

(ii) Exposure for the GCA 6200 stepper

$$\mathrm{M}^2 = \mathrm{M}^1 - 0.64 - 0.000909\mathrm{D} + 0.0000112\mathrm{T}_{res} + \varepsilon_2 \tag{47}$$

where D represents the exposure dose (mJ/cm²). $\varepsilon_2 \sim \mathrm{N}(0, 0.012^2)$

A total of 100 courses simulated from the above system models

Table 1. Percentage of variance captured by each component in example 1

Percent variance captured by MPLS model				
	-----X-Block-----		-----Y-Block-----	
LV #(R)	This LV	Total	This LV	Total
1	25.52	25.52	88.26	88.26
2	26.01	51.53	9.47	97.73
3	23.47	75	1.14	98.87
4	25	100	0.01	98.87

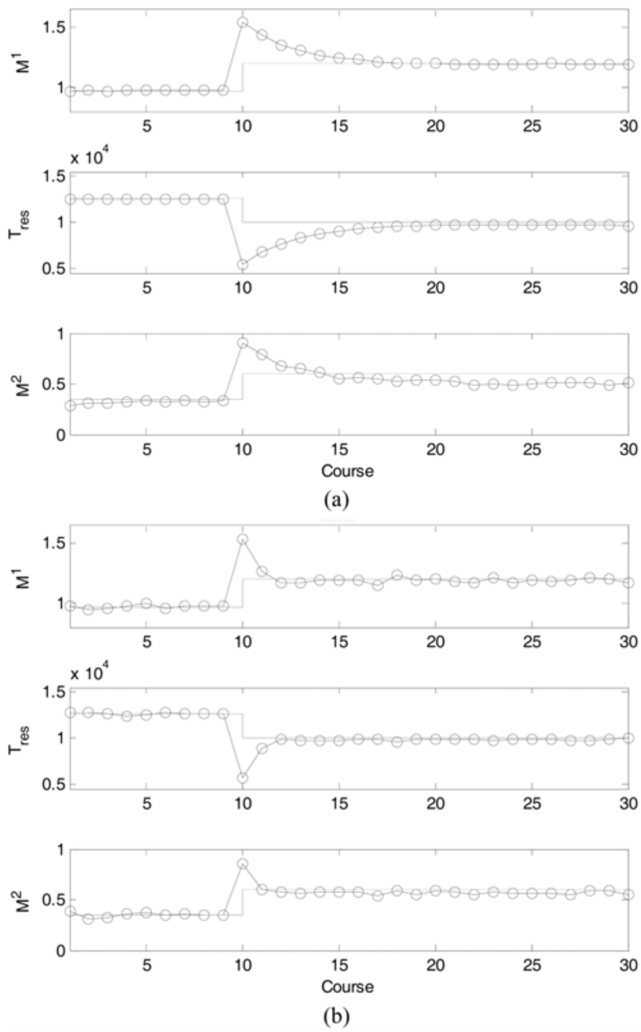

Fig. 5. Convergence of M^1, T_{res} and M^2 for the setpoint change at course 10 in example 1: (a) MBPLS based CtC control, (b) MBPLS based WS control.

is used as the base analysis based on the normal operation whose design quality is set to be T_{res}=12630, M^1=0.97 and M^2=0.4. The operation data is used to build the MBPLS model, in which $\mathbf{x}_i$=[BTE_i SPS_i BTI_i D_i] and $\mathbf{y}_i$=[M_i^1 $T_{res,i}$ M_i^2]. Another 50 courses which do not come from the training sets are produced in a similar simulation for cross validation. The percentage of variance captured by each latent variable is listed in Table 1. The MBPLS model with two latent variables captures over 97.7% of the variance in the in-out relationships of the whole system. To investigate the performance of the proposed method, two cases, including the setpoint change occurring at a stage and the disturbance change occurring at course-to-course, are tested respectively.

(i) Setpoint change at a stage

In the operation of the semiconductor process, the setpoint is often changed to meet the customer requirement. Now the desired qualities at course 10 are changed from M^1=0.97, T_{res}=12610 and M^2= 0.35 to M^1=1.2, T_{res}=10000 and M^2=0.6. The results of the CtC and WS control strategies are shown in Fig. 5(a) and 5(b). Both figures indicate that the control objectives of the two control strategies have been improved gradually and they converge to the desired levels. However, the use of the delayed analysis in CtC results to adjust the multistage process may cause major upset from their setpoints. WS can track the setpoint change more quickly. The corresponding evolution of the control variables of these two control strategies is shown in Fig. 6.

(ii) Disturbance change at two stages

In this case study, a disturbance at stage 1 occurs just from course 10 to 12, where the quality variables M^1 and T_{res} at the first stage are shifted when the unmeasured disturbances with the size of 0.3 and 900 are added, respectively. Also, the average drifts (δ_1^1=0.005 and δ_2^1=40 for stage 1 and δ^2=0.025 for stage 2) per course are included to represent the gradually deactivated qualities,

Stage 1

$$M^1=0.91+1.61\times10^{-3}BTE-2.10\times10^{-5}SPS+\delta_1^1 i+\varepsilon_{11}$$

$$T_{res}=1291.98+928233(SPS)^{-0.5}-1.62BTI-19.49BTE+\delta_2^1 i+\varepsilon_{12} \tag{48}$$

and

$$M^2=M^1-0.64-0.000909D+0.0000112T_{res}+\delta^2 i+\varepsilon_2 \tag{49}$$

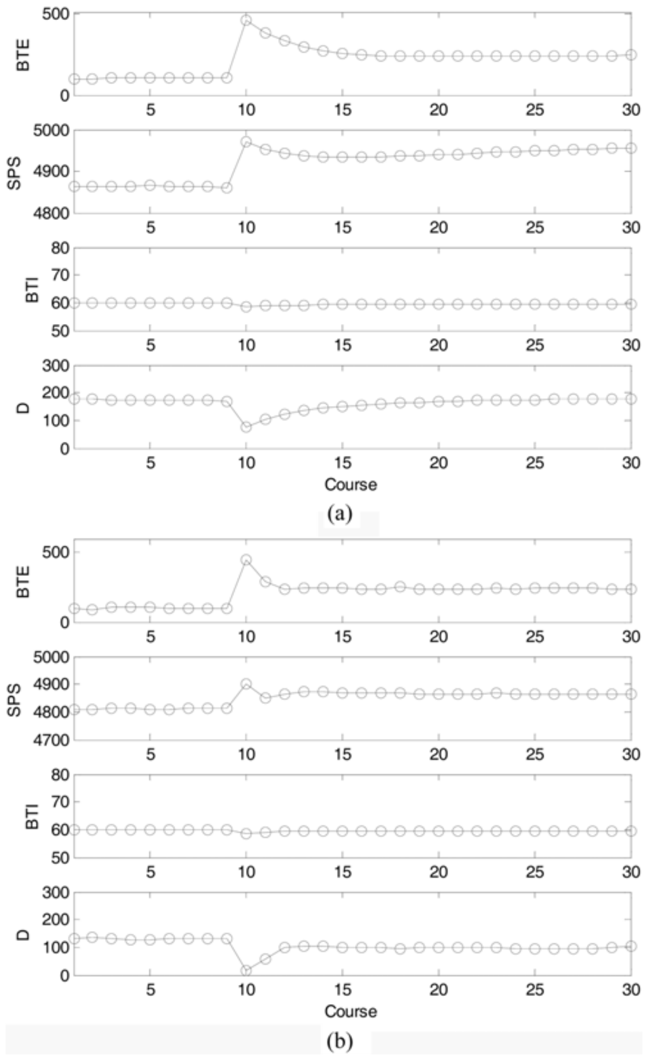

Fig. 6. Designed manipulated variables for the setpoint change at course 10 in example 1: (a) MBPLS based CtC control, (b) MBPLS based WS control.

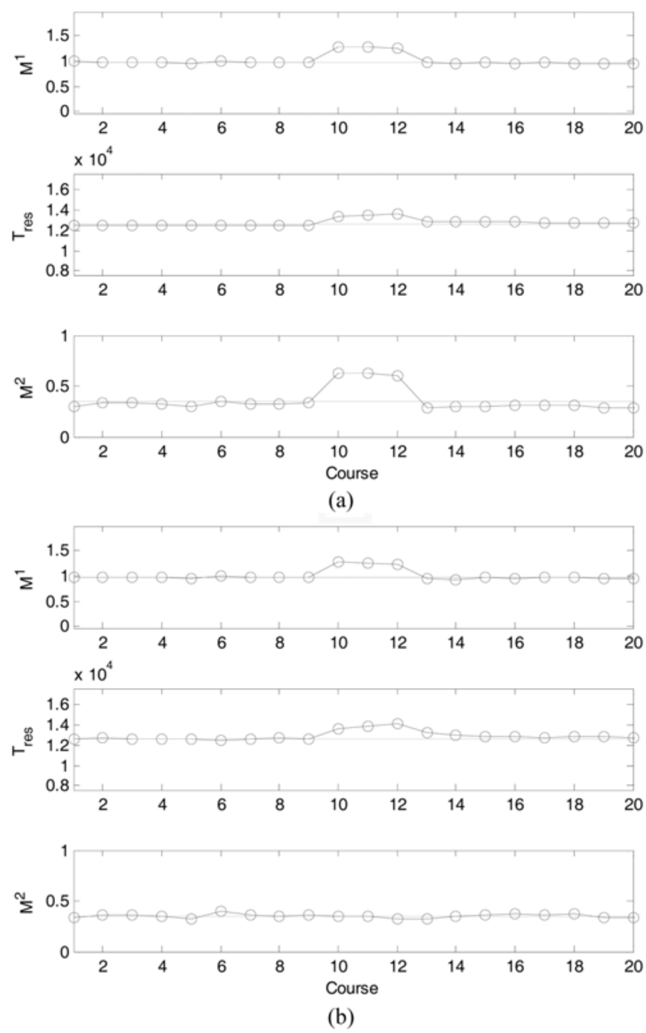

Fig. 7. Convergence of M^1, T_{res} and M^2 for the disturbance change from course 10 to 12 in Example 1: (a) MBPLS based CtC control, (b) MBPLS based WS control.

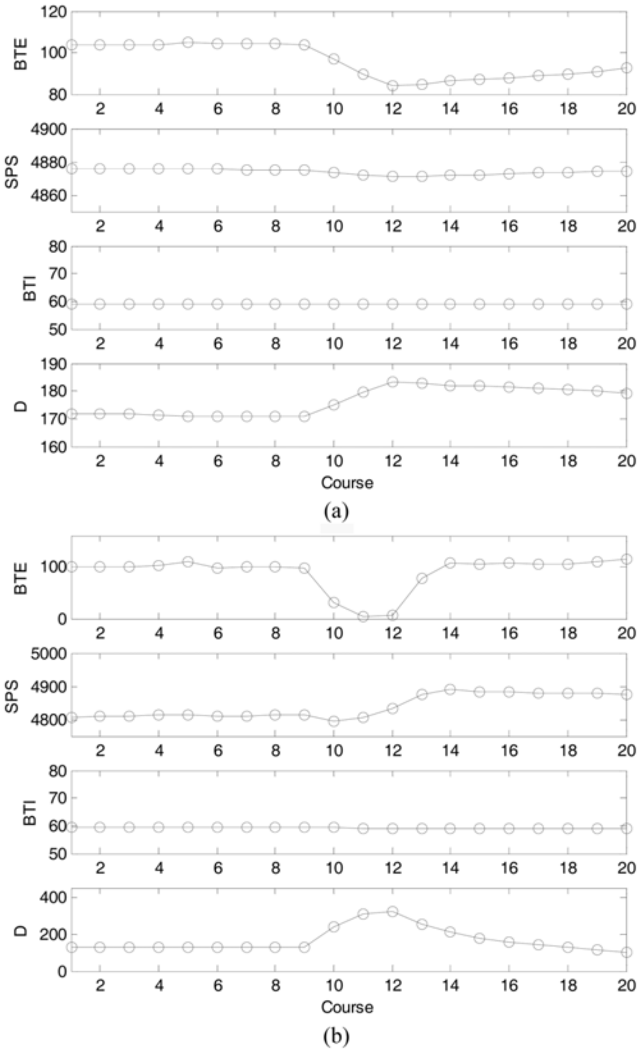

Fig. 8. Designed manipulated variables for the disturbance change from Course 10 to 12 in example 1: (a) MBPLS based CtC control, (b) MBPLS based WS control.

The proposed methods are used to retune the operating condition at each stage operating from course to course. The results of the CtC and WS control strategies are shown in Fig. 7(a) and 7(b). The corresponding manipulated changes are shown in Fig. 8(a) and 8(b). The desired quality at each stage is gradually achieved by WS control before course 14. As shown in Fig. 7(b), the possible improvement can be made by adjusting the operating condition to remove the disturbance affecting the next stage. In this way, the problems of the product qualities are reduced and they will not propagate to the next stage. The WS control strategy can track the disturbance change in a timely manner and maintain the product specifications at an early time point.

2. Example 2: Reverse Osmosis Seawater Desalination

The use of reverse osmosis (RO) membranes in seawater desalination is an established technology. In the reverse osmosis desalination plant, several different stages are often used to get the required fresh water. In this typical example, a series of RO processes consists of four stages with constant duration as shown in Fig. 9. One stage starts as soon as the previous one is finished. In each stage

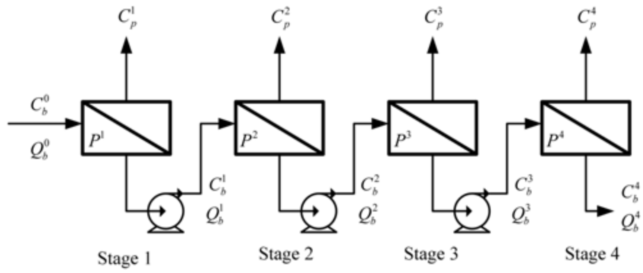

Fig. 9. Four-stage RO configuration.

(h), the composition of salt (C_p^h) in the permeate can be expressed as

$$C_p^h = S^h(C_b^{h-1}, Q_b^{h-1}, P^h) \tag{50}$$

where Q, C and P are the flowrate, concentration, and pressure, respectively. The subscripts b and p denote the brine stream and the permeate stream, respectively. The RO module model (S^h) at each stage h is described as

$$Q_b^{h-1} = Q_b^h + Q_p^h \tag{51}$$

$$Q_b^{h-1} C_b^{h-1} = Q_b^h C_b^h + Q_p^h C_p^h \tag{52}$$

$$Q_p^h = A^h D^h (P^h - \Pi^h) \tag{53}$$

$$C_p^h = \frac{B^h (C_{avg}^h - C_p^h)}{A^h (P^h - \Pi^h)} \tag{54}$$

Eqs. (51) and (52) are the material balance relationships for single membrane unit. Eqs. (53) and (54) represent the flux and the composition of salt for the components passing through the membrane based on a mass transfer solution-diffusion model [19]. Q_p^h is the flow rate of the permeate stream. D is the membrane module area, and P is the operation trans-membrane pressure. C_{avg}^h is the average bulk concentration of the feed side ($C_{avg}^s = (C_b^{s-1} + C_b^s)/2$)). Π is the trans-membrane osmotic pressure,

$$\Pi^h = \frac{2.641 \times 10^5 C_b^{h-1} T}{1 \times 10^6 - C_b^{h-1}} \tag{55}$$

where the operating temperature (T) is 25 °C. A is the water permeability and B is the solute transport parameter. A and B represent the fouling schedule in an exponential decay fashion,

$$A^h = A_0 \exp\left(-\frac{t}{\Gamma_A}\right) \tag{56}$$

$$B^h = B_0 \exp\left(-\frac{t}{\Gamma_B}\right) \tag{57}$$

where A_0 and B_0 denote initial water permeability and the solute transport parameter respectively. Γ_A and Γ_B are the membrane decay constants. t is the operating time. In the above operation, DuPont B-10 (6840) low capacity hollow fiber type of module is considered here. The parameters for calculation, which are given based on Refs [19,20], are listed in Table 2.

One of the main advantages of this sequence operation is flexibility, which is derived from the possibility of readjusting the operation condition of different stages when the membrane module performance deteriorates with the operating time. The proposed meth-

Table 2. Input data for the seawater desalination example

Feed composition, C_b^0	0.0348
Initial water permeability, A_0	3.0×10^{-10}
Initial solute transport parameter, B_0	4.0×10^{-6}
Membrane decay constant, Γ_1	328
Membrane decay constant, Γ_2	650

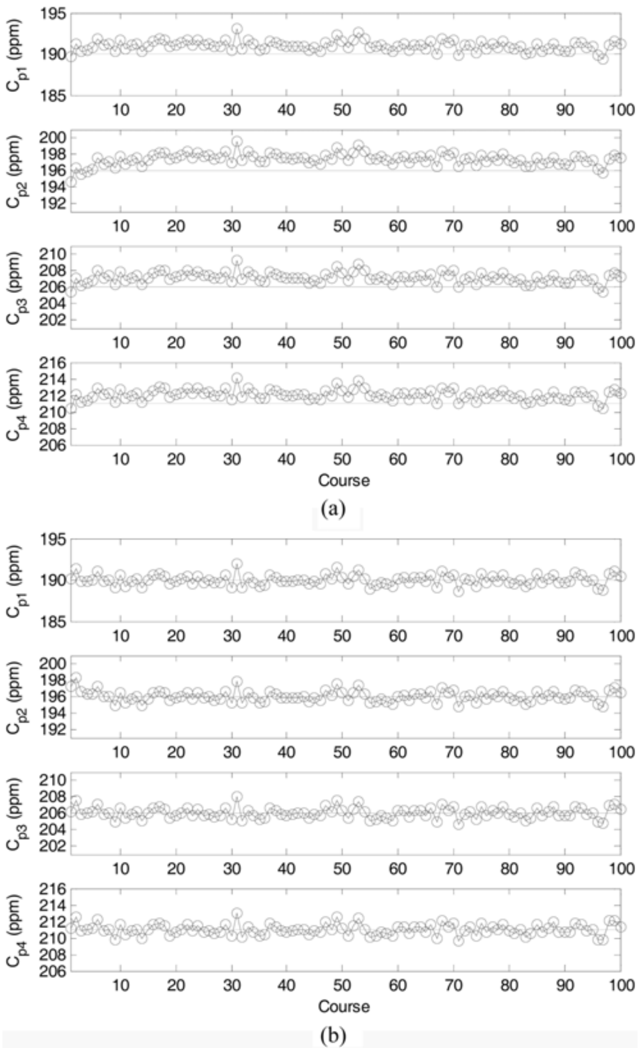

Fig. 10. Convergence of C_p^h, h=1, 2, 3, 4 for the aging effect in example 2: (a) MBPLS-based CtC control, (b) MBPLS-based WS control.

odology is applied to the RO desalination process to maintain the concentration of salt at its desired level of each stage. The feed concentration is 348 ppm, and small variations are added to the input data since there are process variations in the operating system. **y** consists of the measured quality of the composition of salt (C_b^h) in the permeate at each stage, $\mathbf{y}=[y^1\ y^2\ y^3\ y^4]=[C_b^1\ C_b^2\ C_b^3\ C_b^4]$. During each course, the specified composition of salt at each stage is $\mathbf{y}^{sp}=[y^{sp,1}\ y^{sp,2}\ y^{sp,3}\ y^{sp,4}]=[190\ 196\ 206\ 211]$. The manipulated variable **x** is adjusted at each stage for getting the desired qualities, $\mathbf{x}=[x^1\ x^2\ x^3\ x^4]=[P^1\ P^2\ P^3\ P^4]$.

First, we aim to build the MBPLS model. The identification data

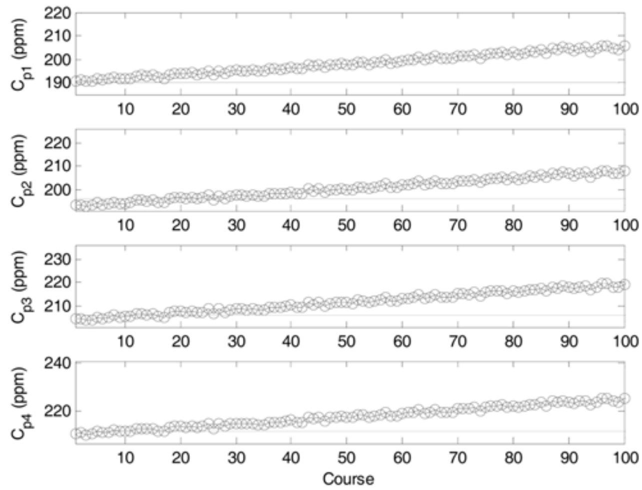

Fig. 11. Convergence of C_p^h, h=1, 2, 3, 4 for the aging effect in example 2 using MBPLS-based WS control without updating the model.

set contains 50 courses based on Eqs. (51)-(57). Another 25 courses, which do not come from the training sets, are produced in a similar way for cross validation. With the cross-validation, four principal components which capture over 87.11% of the variance in the relationships of the multistage process are selected. The proposed methods are used to retune the operating condition at each stage operating from course to course. Fig. 10 depicts the results of the product quality at each stage during each course using the two proposed strategies. The control actions are corrected every four hours. As shown in Fig. 10(a), in the case of MBPLS-based CtC control, the aging effect of the rate constant results in larger errors of the product qualities at the first stage. Furthermore, the disturbance propagates into the rest of the three stages, because the bad performance can only be corrected after the whole course is completely finished. In the case of MBPLS-based WS control (Fig. 10(b)), the disturbances have been significantly reduced at the next stage during the operating course, because WS control can quickly change the designed variables for the rest of the stages instead of the fixed variables of MBPLS-based CtC control. To understand why the performance cannot be improved when the model is not updated, the WS MBPLS-based control design without the updated model is tested. With the plant-model errors and difficulty in control, the final controlled qualities at each stage cannot be converged as shown in Fig. 11.

CONCLUSION

Multiple stage operations are becoming more and more important in manufacturing processes. In this paper, based on the MBPLS modeling structure, two control and optimization methods are developed for serially connected multistage processes. The proposed method has the following advantages for the multivariable serially connected processes.

(1) The MBPLS model, like the conventional PLS model, can extract the strongest relationship between the input and the output variables. It is particularly useful for inherent noise suppression when the unknown disturbance propagates across the whole stage during one course run. Moreover, without the whole system variables, the

conventional SISO design strategy can be directly and separately applied to each pair of the input and the output in the reduced space.

(2) In the conventional approach, the hierarchical decomposition of the serially connected processes into several local sub-processes is often used to reduce the computational burden. The obvious drawback is that only the local relationship between the neighbor sub-processes can be kept instead of the whole stage process. However, MBPLS can still cover the whole stage course even if the stage-by-stage based information is applied instead of the whole-stage one.

(3) The MBPLS based multistage controller is an extension of the single-stage iterative learning optimal design. CtC control, like feedforward control, uses the performance of the previous courses to improve the performance of the operations of the future courses.

(4) As the WS optimal strategy is implemented onto the revolving course, the control action for the rest of the stages can be re-designed without going through all the stages. This allows rejection or reduction of the disturbances just in time before the disturbances affect the rest of the stages, resulting in the disturbances having little effect on the final output quality.

With the robust tracking performance in the presence of exogenous disturbances, simulation experiments illustrate the effectiveness of our proposed approach, which is in accordance with the theoretical conclusion. However, only the noninteracting serial structure is studied in this paper; the work of the interacting serial structure can be extended. Also, the proposed modeling and control method will be expanded to the real applications in our next study.

ACKOWLEDGMENT

The authors wish to express their sincere gratitude to the Center-of-Excellence (COE) Program on Membrane Technology from the Ministry of Education (MOE), R.O.C., to the project Toward Sustainable Green Technology in the Chung Yuan Christian University, Taiwan, under grant CYCU-97-CR-CE, and the National Science Council (NSC) for their financial support.

REFERENCES

1. J. C. Morud and S. Skogestad, *The dynamic behavior of cascade processes with application to distillation columns*, AIChE Annual Meeting, Miami Beach (1995).
2. J. S. Fenner, M. K. Jeong and J.-C. Lu, *IEEE Trans. Semicond. Manuf.*, **18**, 94 (2005).
3. S. Vaidyanathan, *Stochastic control of sequential manufacturing process*, Ph.D dissertation, Carnegie Mellon Univ., Pittsburgh, PA (1991).
4. O. Abel and W. Marquardt, *AIChE J.*, **46**, 803 (2000).
5. Y. Zhang and S. Li, *J. Process Contr.*, **17**, 37 (2007).
6. J. Chen and F. Wang, *J. Process Contr.*, **17**, 309 (2007).
7. J. Flores-Cerrillo and J. F. MacGregor, *Ind. Eng. Chem. Res.*, **44**, 9146 (2005).
8. J. A. Westerhuis and A. K. Smilde, *J. Chemometrics*, **15**, 485 (2001).
9. S. W. Choi and I.-B. Lee, *J. Process Contr.*, **15**, 295 (2005).
10. J. A. Lopes, J. C. Menezes, J. A. Westerhuis and A. K. Smilde, *Biotechnol. Bioeng.*, **80**, 419 (2002).
11. J. F. MacGregor, C. Jaeckle, C. Kiparissides and M. Kourtoudi, *AIChE J.*, **40**, 826 (1994).
12. S. J. Qin, S. Valle and M. J. Piovoso, *J. Chemometrics*, **15**, 715 (2001).
13. W. R. Zwick and W. F. Velicer, *Psychological Bulletin*, **99**, 432 (1986).
14. S. W. Choi and I.-B. Lee, *J. Process Contr.*, **15**, 295 (2005).
15. S. Wold, N. Kettaneh and K. Tjessem, *J. Chemometrics*, **10**, 463 (1996).
16. L. E. Wangen and B. R. Kowalski, *J. Chemometrics*, **3**, 3 (1988).
17. S. Leang, S. Y. Ma, J. Thomson, B. J. Bombay and C. J. Spanos, *IEEE Trans. Semicond. Manuf.*, **9**, 191 (1996).
18. S. Leang and C. J. Spanos, *IEEE Trans. Semicond. Manuf.*, **9**, 101 (1996).
19. Y.-Y. Lu, Y.-D. Hu, D.-M. Xu and L.-Y. Wu, *J. Membr. Sci.*, **282**, 7 (2006).
20. M. Zhu, M. M. El-Halwagi and M. Al-Ahmad, *J. Membr. Sci.*, **129**, 161 (1997).