Proceedings of the International MultiConference of Engineers and Computer Scientists 2008 Vol II
IMECS 2008, 19-21 March, 2008, Hong Kong

Two-Step MPLS-Based Iterative Learning Control for Batch Processes

J. Chen, *Member, IAENG* and K.-C. Lin

Abstract—**A novel iterative learning control scheme is proposed for control of the end point quality in batch processes. This new method is based on two-step multi-way partial least squares (MPLS) models. The first-step MPLS model, called the quality MPLS, is used to relate the final qualities with the on-line measurements; the second one, called the measurement MPLS, can relate the on-line measurements with both the manipulated variables and the prior measured variables. Because the standard MPLS model embeds all process variables into a single input data block, the input variables can be adjusted based on target qualities at the end point. The adjusted inputs in the latent model are fixed when on-line batch control is applied. This will hinder the system's quick response from the required changes at each time point. With the proposed method, the desired on-line measured variables along the time axis can be computed using the quality MPLS; then the operating input variables in the latent model can be appropriately adjusted using the measurement MPLS to match the desired on-line measured variables at each time point and finally get the target qualities at the end time point. The applications are discussed through a typical batch reactor to demonstrate the advantages of the proposed method in comparison with the conventional methods.**

Index Terms— **Batch Processes; Iterative Learning Control; Multiway Partial Least Squares; Optimization**

I. Introduction

The need to efficiently operate the complexity of industrial semi-batch or batch processes has motivated the development of the new control techniques over the past few years. This especially rings true in the batch processes mainly involved in the production of high-value-added specialty chemicals, such as polymer reaction systems, pharmaceutical manufactures, biochemical and semiconductor processes. The operation of the batch process is quite different from that of the continuous process, because the former, as the name indicates, is characterized by prescribed processing of raw materials as a discrete entity for finite amount of time when producing a finished product. It is transient in nature with their state changed with time so that the operating trajectories of the manipulated variables can have significant effects on the final product characteristics.

The control design of batch process has both dynamic and static constraints, so it falls into the class of dynamic optimization problem [1]. When the parameters are associated with the models that are accurate and truly representative of the original batch process, the mathematical model can be used in the optimization methods [2]. However, such accurate models are seldom available in reality, especially for batch processes. Parameters involved with modeling, kinetics and thermodynamics of a reaction, for instance, are not completely established for most of the batch processes. Their dynamic nature is also not very well understood. Some research in the past focused on building empirical models, such as neural networks and fuzzy logic, because of rich information of the operating data. Zhao, et al. used a multi-layer feedforward neural network to model the batch polypropylene reactor and controlled it by an intelligent compound system consisted of bang-bang control, fuzzy control and traditional PID control [3]. A trained feedforward artificial neural network was proposed to estimate the heat generation term and control the temperature by the feedforward/feedback control algorithm [4]. The batch-to-batch iterative control strategy based on the neural network model was also proposed [5]. Their trained neural network model was refined using the data from the previous batch run. However, the nonlinear predictive control design based on neural network models was computationally demanding.

Recently, data-driven multivariable statistical techniques, such as multi-way principal component analysis and multi-way partial least squares (MPLS), have been widely used to handle high dimensional and coupled data. They attracted lots of interest in involving profile tracking control in batch processes [6, 7]. Because the standard MPLS model embeds the entire batch data in a single data block, the adjusted input variables in the latent space cannot be changed during each batch run. This will hinder the system's quick response from fixing the disturbance at each time point. In this paper, the two-step MPLS models are developed to reflect the direct correlation between manipulated variables and measured variables and to improve the performance of the batch process monitoring and quality control. The scheme first divides the process data into two parts. In the first part, a quality MPLS model is constructed to relate the final qualities with the on-line measurements; a measurement MPLS model is used for the second part to relate the on-line measurements with both the manipulated variables and the prior measured variables. With the quality MPLS, the desired on-line measured variables at each time point can be estimated from

This work was sponsored in part by National Science Council, R.O.C., and in part by the Ministry of Economic, R.O.C.
J. Chen is with R & D Center for Membrane Technology, Department of Chemical Engineering, Chung-Yuan Christian University, Chung-Li, Taiwan 320, Republic of China (corresponding author to provide fax: 886-3-265-4199; e-mail: jason@ wavenet.cycu.edu.tw).
K.-C.. Lin was with Department of Chemical Engineering, Chung-Yuan Christian University, Chung-Li, Taiwan 320, Republic of China. He is now with Nanya Technology Corp., Hwa-Ya Technology Park, Taoyuan, Taiwan, Republic of China.

the end point target qualities. The measurement MPLS can then appropriately adjust the manipulated variable profiles to reach the desired on-line measured variables at each time point along the time axis. Because of the model-plant mismatches and unmeasured disturbances, the calculated optimal trajectories may not be optimal when they are implemented onto the actual batch unit. An iterative learning control method is developed to calculate and adjust a recipe.

The remainder of this paper is organized as follows: The second section defines the design problem of the end point quality control. The two-step MPLS models are developed in Section III. In Section IV, the batch-to-batch control is derived based on the two-step MPLS models. The effectiveness of the proposed method is demonstrated through a simulation benchmark of a batch reactor in Section V. This example investigates the performance of the proposed method. A comparison with the conventional algorithms is also made. Finally, concluding remarks are made.

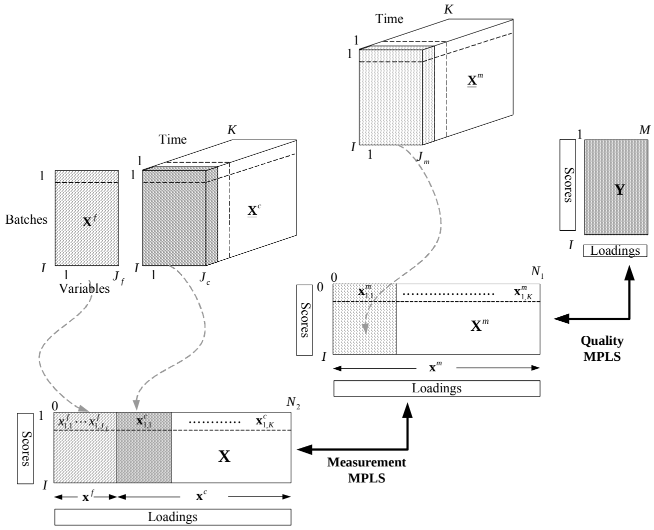


Fig. 1. Data structure for modeling a batch process.

II. End-Point Quality Control

The batch control design aims at seeking appropriate manipulated variables ($\mathbf{x}^c$) to make the final product qualities ($\mathbf{y}(t_f)$) match the desired qualities ($\mathbf{y}^{sp}$) when the batch is finished. It can be mathematically formulated as

$$J = \frac{1}{2}\min_{\mathbf{x}^c}\left\|\mathbf{y}^{sp} - \mathbf{y}(t_f)\right\| \tag{1}$$

subject to

$$\begin{aligned} &\dot{\mathbf{x}}^m = \mathbf{g}(\mathbf{x}^m, \mathbf{x}^f, \mathbf{x}^c), \qquad 0 \le t \le t_f \\ &\mathbf{y} = \mathbf{h}(\mathbf{x}^m, \mathbf{x}^f, \mathbf{x}^c) \\ &\qquad \text{and} \quad \mathbf{L} \le \mathbf{x}^c(t) \le \mathbf{U} \end{aligned} \tag{2}$$

where $\mathbf{y}(1\times M)$ and $\mathbf{y}^{sp}(1\times M)$ are the actual and the desired product of M quality variables respectively. However, $\mathbf{y}(t_f)(1\times M)$ can only be measured after the batch run is finished. t_f is the duration of each batch run. $\mathbf{x}^m(1\times J_m)$ is a vector with J_m on-line measured variables at each sampling point. $\mathbf{x}^f(1\times J_f)$ is a vector with J_f off-line collected measurements before running the batch. $\mathbf{x}^c(1\times J_c)$ is J_c profiles of the manipulated variables at each sampling point. $\mathbf{U}$ and $\mathbf{L}$ are the lower and upper bounds of the manipulated trajectory, respectively. Although $\mathbf{g}$ and $\mathbf{h}$ give the description of the dynamic model of the batch process, it is highly difficult to get a reliable process model. Here a data-driven approach is used to construct the model. When there are more than one manipulated variable ($J_c > 1$) and quality variable ($M > 1$), a set of I number of historical batch data over K time intervals can be constructed into three-way arrays and two-way arrays shown in Fig. 1. The two two-way data matrices $\mathbf{X}^f(I\times J_f)$ and $\mathbf{Y}(I\times M)$ are defined. The two three-way data matrices are $\underline{\mathbf{X}}^m(I\times J_m\times K)$ and $\underline{\mathbf{X}}^c(I\times J_c\times K)$.

III. Two-step MPLS Modeling

The given process data are divided into three blocks, a quality block ($\mathbf{Y}$), an on-line measured block ($\mathbf{X}^m$) and a manipulated block ($\mathbf{X}$). $\mathbf{Y}$ block with a two-way array ($I\times M$) summarizes the I runs and the M final properties (or responses). $\mathbf{X}^m(I\times N_1)$ block is constructed by unfolding $\underline{\mathbf{X}}^m(I\times J_m\times K)$ shown in Fig. 1, where $\mathbf{x}^m(1\times J_m)$ at each time point are separately filled in the rest of columns of the $\mathbf{X}^m$ block along the time axis and $N_1 = J_m K$. $\mathbf{X}$ block with a two-way array ($I\times N_2$) organizes $\mathbf{X}^f(I\times J_f)$ and $\underline{\mathbf{X}}^c(I\times J_c\times K)$, where $N_2 = J_f + J_c K$. In Fig. 1, the first J_f columns of the $\mathbf{X}$ block are the off-line measured data. $\mathbf{x}^c(1\times J_c)$ at each time point are separately filled in the rest of columns of the $\mathbf{X}$ block along the time axis. After $\mathbf{X}^m$, $\mathbf{X}$ and $\mathbf{Y}$ are scaled and mean centered, similar as in the standard MPLS, the MPLS algorithm is separately applied to { $\mathbf{X}$, $\mathbf{X}^m$ } and { $\mathbf{X}^m$, $\mathbf{Y}$ }. The two outer models are:

Quality MPLS

$$\begin{aligned} \mathbf{X}^m &= \mathbf{T}^q\left(\mathbf{W}^q\right)^T + \mathbf{E}^q \\ \mathbf{Y} &= \mathbf{U}^q\left(\mathbf{Q}^q\right)^T + \mathbf{F}^q \end{aligned} \tag{3}$$

and Measurement MPLS

$$\begin{aligned} \mathbf{X} &= \mathbf{T}^m\left(\mathbf{W}^m\right)^T + \mathbf{E}^m \\ \mathbf{X}^m &= \mathbf{U}^m\left(\mathbf{Q}^m\right)^T + \mathbf{F}^m \end{aligned} \tag{4}$$

where

$$\mathbf{U}^q = \begin{bmatrix} u^q_{1,1} & \cdots & u^q_{1,r} & \cdots & u^q_{1,R_q} \\ \vdots & & \vdots & & \vdots \\ u^q_{i,1} & \cdots & u^q_{i,r} & \cdots & u^q_{i,R_q} \\ \vdots & & \vdots & & \vdots \\ u^q_{I,1} & \cdots & u^q_{I,r} & \cdots & u^q_{I,R_q} \end{bmatrix} = \begin{bmatrix} \left(\mathbf{u}^q_1\right)^T \\ \vdots \\ \left(\mathbf{u}^q_i\right)^T \\ \vdots \\ \left(\mathbf{u}^q_I\right)^T \end{bmatrix} \tag{5}$$

$$\mathbf{U}^m = \begin{bmatrix} u^m_{1,1} & \cdots & u^m_{1,r} & \cdots & u^m_{1,R_m} \\ \vdots & & \vdots & & \vdots \\ u^m_{i,1} & \cdots & u^m_{i,r} & \cdots & u^m_{i,R_m} \\ \vdots & & \vdots & & \vdots \\ u^m_{I,1} & \cdots & u^m_{I,r} & \cdots & u^m_{I,R_m} \end{bmatrix} = \begin{bmatrix} (\mathbf{u}^m_1)^T \\ \vdots \\ (\mathbf{u}^m_i)^T \\ \vdots \\ (\mathbf{u}^m_I)^T \end{bmatrix} \tag{6}$$

$$\mathbf{T}^q = \begin{bmatrix} t^q_{1,1} & \cdots & t^q_{1,r} & \cdots & t^q_{1,R_q} \\ \vdots & & \vdots & & \vdots \\ t^q_{i,1} & \cdots & t^q_{i,r} & \cdots & t^q_{i,R_q} \\ \vdots & & \vdots & & \vdots \\ t^q_{I,1} & \cdots & t^q_{I,r} & \cdots & t^q_{I,R_q} \end{bmatrix} = \begin{bmatrix} (\mathbf{t}^q_1)^T \\ \vdots \\ (\mathbf{t}^q_i)^T \\ \vdots \\ (\mathbf{t}^q_I)^T \end{bmatrix} \tag{7}$$

$$\mathbf{T}^m = \begin{bmatrix} t^m_{1,1} & \cdots & t^m_{1,r} & \cdots & t^m_{1,R_m} \\ \vdots & & \vdots & & \vdots \\ t^m_{i,1} & \cdots & t^m_{i,r} & \cdots & t^m_{i,R_m} \\ \vdots & & \vdots & & \vdots \\ t^m_{I,1} & \cdots & t^m_{I,r} & \cdots & t^m_{I,R_m} \end{bmatrix} = \begin{bmatrix} (\mathbf{t}^m_1)^T \\ \vdots \\ (\mathbf{t}^m_i)^T \\ \vdots \\ (\mathbf{t}^m_I)^T \end{bmatrix} \tag{8}$$

$\mathbf{W}^q = \lfloor \mathbf{w}^q_1 \ \cdots \ w^q_{R_q} \rfloor$ and $\mathbf{Q}^q = \lfloor \mathbf{q}^q_1 \ \cdots \ \mathbf{q}^q_{R_q} \rfloor$ are the loading matrices which show the influence of $\mathbf{X}^m$ and $\mathbf{Y}$. $\mathbf{E}^q$ and $\mathbf{F}^q$ are the residual matrices that are not useful for describing $\mathbf{X}^m$ and $\mathbf{Y}$ respectively. For the measurement MPLS model, similar definitions can be used to explain the relationship between $\mathbf{X}$ and $\mathbf{X}^m$. R_q and R_m are the number of principal components. The inner relationships between $\mathbf{X}^m$ and $\mathbf{Y}$ and between $\mathbf{X}$ and $\mathbf{X}^m$ are separately given as:

Quality MPLS

$$\mathbf{U}^q = \mathbf{B}^q \mathbf{T}^q \tag{9}$$

Measurement MPLS

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

$\mathbf{B}^q = diag(b^q_1 \ \cdots \ b^q_{R_q})$ and $\mathbf{B}^m = diag(b^m_1 \ \cdots \ b^m_{R_m})$ are diagonal matrices containing the regression coefficients (b^q_r and b^m_r) of the score model, respectively. Now, after the MIMO regression process is calculated by MPLS, the data blocks ({ $\mathbf{X}$, $\mathbf{X}^m$ } and { $\mathbf{X}^m$, $\mathbf{Y}$ }) are broken into a series of univariate regression processes (Fig. 2).

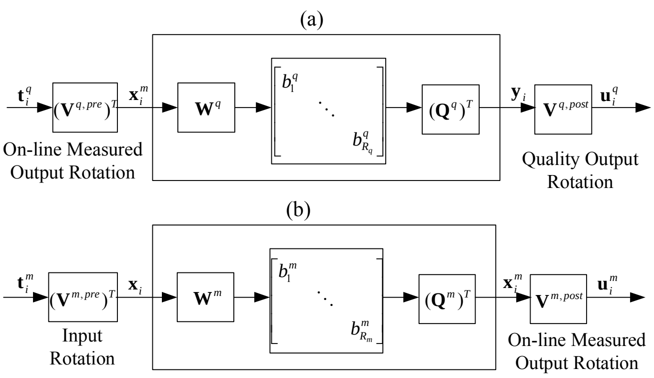


Fig. 2. Diagonalize the batch process: (a) on-line measured outputs and quality outputs; (b) inputs and on-line measured outputs.

In Fig. 2, $\mathbf{V}^{q,post}$ and $\mathbf{V}^{q,pre} = \mathbf{W}^q$ are the post-multiplied and the pre-multiplied operators for $\mathbf{X}^m$ and $\mathbf{Y}$. $\mathbf{V}^{m,post}$ and $\mathbf{V}^{m,pre} = \mathbf{W}^m$ are the post-multiplied and the pre-multiplied operators for $\mathbf{X}$ and $\mathbf{X}^m$. After projection, the transformations of input $\mathbf{x}^m_i = \lfloor \mathbf{x}^m_{i,1} \ \cdots \ \mathbf{x}^m_{i,K} \rfloor$ and output $\mathbf{y}_i = \lfloor y_{i,1} \ \cdots \ y_{i,M} \rfloor$ at the batch run i are expressed respectively,

$$\mathbf{x}^m_i \mathbf{V}^{q,pre} = (\mathbf{t}^q_i)^T, \ \mathbf{y}_i \mathbf{V}^{q,post} = (\mathbf{u}^q_i)^T \tag{11}$$

The projection of output $\mathbf{x}^m_i = \lfloor \mathbf{x}^m_{i,1} \ \cdots \ \mathbf{x}^m_{i,K} \rfloor$ and input $\mathbf{x}_i = \lfloor \mathbf{x}^f_i \ \mathbf{x}^c_{i,1} \ \cdots \ \mathbf{x}^c_{i,K} \rfloor$ at the batch run i can be represented respectively,

$$\mathbf{x}_i \mathbf{V}^{m,pre} = (\mathbf{t}^m_i)^T, \ \mathbf{x}^m_i \mathbf{V}^{m,post} = (\mathbf{u}^m_i)^T \tag{12}$$

As a result, the transformed inputs and outputs for $\mathbf{t}^q_i$ and $\mathbf{u}^q_i$ and for $\mathbf{t}^m_i$ and $\mathbf{u}^m_i$ are completely decoupled (Fig. 2).

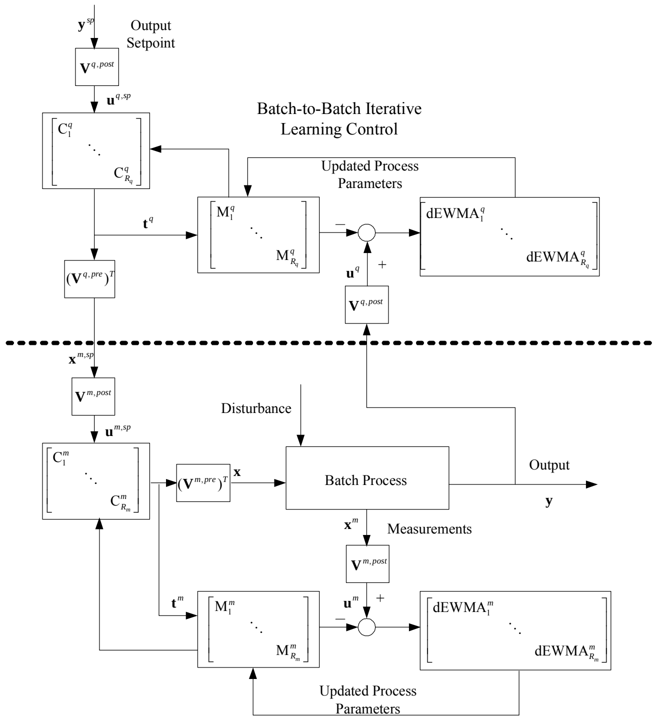


Fig. 3. The structure ILC design using two-step MPLS models.

IV. Two-step MPLS-Based Iterative Learning Control Design

The goal of the iterative learning control (ILC) design is to seek appropriate control input profiles $\mathbf{x}^c_i$ that yield the product quality $\mathbf{y}_i$ at the next batch run i to match the desired targets $\mathbf{y}^{sp}$. The block diagram of the two-step MPLS based ILC system is shown in Fig. 3.

A. Computing the Desired On-line Measured Variables using Quality MPLS

$\mathbf{y}^{sp}$ and $\mathbf{y}_i$ are decomposed into the lower dimensional space $\mathbf{y}^{sp} = \mathbf{y}^{sp}_{PLS} + \mathbf{f}^{q,sp}$ and $\mathbf{y}_i = \mathbf{y}_{PLS,i} + \mathbf{f}^q_i$. The end point of the control objective ((1)) can be represented as

$$J^q = \frac{1}{2}\min_{\mathbf{x}_i^m}\left\|\mathbf{y}^{sp} - \mathbf{y}_i\right\|^2 = \frac{1}{2}\min_{\mathbf{x}_i^m}\left\|\mathbf{y}_{PLS}^{sp} - \mathbf{y}_{PLS,i}\right\|^2 \quad (13)$$

Substituting $\mathbf{y}_{PLS}^{sp} = \sum_{r=1}^{R_q} u_r^{q,sp}(\mathbf{q}_r^q)^T$ and $\mathbf{y}_{PLS,i} = \sum_{r=1}^{R_q} u_{i,r}^q(\mathbf{q}_r^q)^T$ into (13),

$$\begin{aligned} J^q &\cong \frac{1}{2}\min_{\mathbf{x}_i^m}\left\|\sum_{r=1}^{R_q}\left(u_r^{q,sp} - u_{i,r}^q\right)(\mathbf{q}_r^q)^T\right\|^2 \\ &\le \left[\min_{t_{i,1}^q} J_1^q + \min_{t_{i,2}^q} J_2^q + \cdots\cdots + \min_{t_{i,R_q}^q} J_{R_q}^q\right] \end{aligned} \quad (14)$$

where $J_r^q \equiv \frac{1}{2}\left(u_r^{q,sp} - u_{i,r}^q\right)^2\left\|(\mathbf{q}_r^q)^T\right\|^2$. The objective function is decomposed into R_q sub-objective functions in the lower dimensional subspace, $J^q = \sum_{r=1}^{R_q} J_r^q$. Because of the persistent disturbance, the developed quality MPLS model ($u_{i,r}^q = b_r^q t_{i,r}^q$) does not always have accurate correlation or the process model is subject to changes of the disturbances in the process. Here the observer is assumed that each output score ($\hat{\mathbf{u}}_i^q$) is the linear combination of the corresponding input score ($\mathbf{t}_i^q$), a deterministic offset ($\mathbf{a}_i^q$) and a deterministic trend disturbance ($\mathbf{d}_i^q$),

$$\begin{aligned} \hat{\mathbf{u}}_i^q = \begin{bmatrix} \hat{u}_{i,1}^q \\ \vdots \\ \hat{u}_{i,R_q}^q \end{bmatrix} &= \begin{bmatrix} a_{i,1}^q \\ \vdots \\ a_{i,R_q}^q \end{bmatrix} + \begin{bmatrix} b_1^q & 0 & 0 \\ 0 & \ddots & 0 \\ 0 & 0 & b_{R_q}^q \end{bmatrix}\begin{bmatrix} t_{i,1}^q \\ \vdots \\ t_{i,R_q}^q \end{bmatrix} + \begin{bmatrix} d_{i,1}^q \\ \vdots \\ d_{i,R_q}^q \end{bmatrix} \\ &= \mathbf{a}_i^q + \mathbf{B}^q\mathbf{t}_i^q + \mathbf{d}_i^q \end{aligned} \quad (15)$$

This equation represents the model $\mathbf{M}^q = \left\{\mathrm{M}_r^q\right\}_{r=1,\cdots,R_q}$ shown in Fig. 3. The bias term $\mathbf{a}_i^q$ and the trend estimation filter $\mathbf{d}_i^q$ are recursively updated using the traditional dEWMA method. Thus, the adjusting the input score factor at the value

$$\mathbf{t}_i^q = \begin{bmatrix} t_{i,1}^q \\ \vdots \\ t_{i,R_q}^q \end{bmatrix} = (\mathbf{B}^q)^{-1}\left(\mathbf{u}^{q,sp} - \mathbf{a}_i^q - \mathbf{d}_i^q\right) \quad (16)$$

This equation represents the controller $\mathbf{C}^q = \left\{\mathrm{C}_r^q\right\}_{r=1,\cdots,R_q}$ shown in Fig. 3. Thus, the desired on-line measured variables ($\mathbf{x}_i^{m,sp}$) for the i batch run are

$$\mathbf{x}_i^{m,sp} = \sum_{r=1}^{R_q} t_{i,r}^q(\mathbf{w}_r^q)^T \quad (17)$$

B. Computing the Desired Manipulated Variables Using Measurement MPLS

Based on the desired $\mathbf{x}_i^{m,sp}$, the new control objective is defined as

$$J^m = \frac{1}{2}\min_{\mathbf{x}_i^c}\left\|\mathbf{x}_i^{m,sp} - \mathbf{x}_i^m\right\|^2 \quad (19)$$

Like the previous procedure, the desired $\mathbf{x}_i^{m,sp}$ and $\mathbf{x}_i^m$ are decomposed into the lower dimensional space

$$J^m = \frac{1}{2}\min_{\mathbf{x}_i^c}\left\|\mathbf{x}_i^{m,sp} - \mathbf{x}_i^m\right\| \cong \frac{1}{2}\min_{\mathbf{x}_i^c}\left\|\mathbf{x}_{PLS,i}^{m,sp} - \mathbf{x}_{PLS,i}^m\right\|^2 \quad (20)$$

Substituting $\mathbf{x}_{PLS,i}^{m,sp} = \sum_{r=1}^{R_m} u_{i,r}^{m,sp}(\mathbf{q}_r^m)^T$ and $\mathbf{x}_{PLS,i}^m = \sum_{r=1}^{R_m} u_{i,r}^m(\mathbf{q}_r^m)^T$ into (20),

$$\begin{aligned} J^m &\cong \frac{1}{2}\min_{\mathbf{x}_i^c}\left\|\sum_{r=1}^{R_m}\left(u_{i,r}^{m,sp} - u_{i,r}^m\right)(\mathbf{q}_r^m)^T\right\|^2 \\ &= \left[\min_{t_{i,1}^m} J_1^m + \min_{t_{i,2}^m} J_2^m + \cdots\cdots + \min_{t_{i,R_m}^m} J_{R_m}^m\right] \end{aligned} \quad (21)$$

Assume that each output score ($\hat{\mathbf{u}}_i^m$) is the linear combination of the corresponding input score ($\mathbf{t}_i^m$), a deterministic offset ($\mathbf{a}_i^m$) and a deterministic trend disturbance ($\mathbf{d}_i^m$),

$$\begin{aligned} \hat{\mathbf{u}}_i^m = \begin{bmatrix} \hat{u}_{i,1}^m \\ \vdots \\ \hat{u}_{i,R_m}^m \end{bmatrix} &= \begin{bmatrix} a_{i,1}^m \\ \vdots \\ a_{i,R_m}^m \end{bmatrix} + \begin{bmatrix} b_1^m & 0 & 0 \\ 0 & \ddots & 0 \\ 0 & 0 & b_{R_m}^m \end{bmatrix}\begin{bmatrix} t_{i,1}^m \\ \vdots \\ t_{i,R_m}^m \end{bmatrix} + \begin{bmatrix} d_{i,1}^m \\ \vdots \\ d_{i,R_m}^m \end{bmatrix} \\ &= \mathbf{a}_i^m + \mathbf{B}^m\mathbf{t}_i^m + \mathbf{d}_i^m \end{aligned} \quad (22)$$

This equation represents the model $\mathbf{M}^m = \left\{\mathrm{M}_r^m\right\}_{r=1,\cdots,R_m}$ shown in Fig. 3. The previous bias term $\mathbf{a}_i^m$ and the trend estimation filter $\mathbf{d}_i^m$ are also updated using dEWMA. Thus, the controllable factor that achieves the target value $\mathbf{u}_i^{m,sp} = \left[u_{i,1}^{m,sp} \ \cdots \ u_{i,R_m}^{m,sp}\right]^T$ is

$$\mathbf{t}_i^m = \begin{bmatrix} t_{i,1}^m \\ \vdots \\ t_{i,R_m}^m \end{bmatrix} = (\mathbf{B}^m)^{-1}\left(\mathbf{u}_i^{m,sp} - \mathbf{a}_i^m - \mathbf{d}_i^m\right) \quad (23)$$

This equation represents the controller $\mathbf{C}^m = \left\{\mathrm{C}_r^m\right\}_{r=1,\cdots,R_m}$ shown in Fig. 3. Thus, the desired input profile variables ($\mathbf{x}_i^c$) for the i batch run are

$$\hat{\mathbf{x}}_i = \sum_{r=1}^{R_m} t_{i,r}^m(\mathbf{w}_r^m)^T \quad (24)$$

where $\hat{\mathbf{x}}_i = \left[\hat{\mathbf{x}}_i^f \quad \mathbf{x}_{1,i}^c \quad \cdots \quad \mathbf{x}_{K,i}^c\right]$

C. ILC Control Design

The input profiles $\mathbf{x}_i^c = \left[\mathbf{x}_{i,1}^c \quad \cdots \quad \mathbf{x}_{i,K}^c\right]$ computed from (24) for the current batch run cannot be directly implemented, because the computed $\hat{\mathbf{x}}_i^f$ will not be equal to the actual off-line measured variables $\mathbf{x}_i^f$. The error ($\boldsymbol{\delta}_i^f$) of $\hat{\mathbf{x}}_i^f$ and $\mathbf{x}_i^f$ in the score space is defined as

$$\boldsymbol{\delta}_i^f \equiv \hat{\mathbf{x}}_i^f - \mathbf{x}_i^f \quad (25)$$

In order to achieve the desired target qualities, the computed score value $\mathbf{t}_i^m = \left[t_{i,1}^m \quad \cdots \quad t_{i,R_m}^m\right]^T$ still should be applied. Substituting (25) into (24),

$$\begin{aligned} \left(\mathbf{t}_i^m\right)^T &= \left[\hat{\mathbf{x}}_i^f \quad \hat{\mathbf{x}}_i^c\right]\begin{bmatrix} \mathbf{W}_{1,f}^m \\ \mathbf{W}_{2,f}^m \end{bmatrix} \\ &= \mathbf{x}_i^f\mathbf{W}_{1,f}^m + \boldsymbol{\delta}_i^f\mathbf{W}_{1,f}^m + (\mathbf{t}_i^m)^T(\mathbf{P}_{2,f}^m)^T\mathbf{W}_{2,f}^m \end{aligned} \quad (26)$$

Assume the adjusted error term ($\boldsymbol{\delta}_i^f\mathbf{W}_{1,f}^m$) in the $(\mathbf{P}_{2,f}^m)^T\mathbf{W}_{2,f}^m$ space is defined as

$$\boldsymbol{\delta}_i^f\mathbf{W}_{1,f}^m \equiv \boldsymbol{\alpha}_i^m(\mathbf{P}_{2,f}^m)^T\mathbf{W}_{2,f}^m \quad (27)$$

where $\mathbf{P} = \begin{bmatrix} \mathbf{P}_{1,f}^m \\ \mathbf{P}_{2,f}^m \end{bmatrix}$ is the loading matrix for $\mathbf{X}$. Rewrite (26) and substitute it into (24)

$$\mathbf{x}_i^c = ((\mathbf{t}_i^m)^T + \boldsymbol{\alpha}_i^m)(\mathbf{P}_{2,f}^m)^T = ((\mathbf{t}_i^m)^T - \mathbf{x}_i^f \mathbf{W}_{1,f}^m)((\mathbf{P}_{2,f}^m)^T \mathbf{W}_{2,f}^m)^{-1}(\mathbf{P}_{2,f}^m)^T \tag{28}$$

V. Illustration Examples

This example is intended to show how the proposed control techniques can be used to design an iterative learning batch controller for a typical exothermic chemical batch reactor. The differential equations and the simulation condition describing the reaction process are given by [8]. The reaction system involves two consecutive first-order reactions, $A \to B \to C$. Two stages are run in the system. In the first (start-up) stage, the steam in the jacket initially heats up the reactor content until the exothermic heat of reaction is generated significantly enough. In the second (maintenance) stage, the cooling water in the jacket is used to remove the exothermic heats of reaction. The control structure of the system consists of two levels. In the lower level, the reactor temperature is controlled by two split-ranged control valves, a steam valve and a water valve. The higher level of control is needed to assure the quality of the product of interest at the end of each batch run. Because of the lack of in-situ measurements of the product qualities, the products must be moved from the reactor to a laboratory before an accurate measurement of the control concentration can be taken. Alternatively, other more easily measurable variables, such as the reactor temperature, may be used as an indicator for the compositions of the product. $\mathbf{x}^f$ (initial concentration of C_A) and $\mathbf{x}^m$ (the jacket and the reactor temperatures) are collected during each batch run. $\mathbf{x}^c$ is the designed temperature of the reactor to be computed from the design controller for the desired qualities at the end of the batch run. The control objective is to maintain the concentrations of A and B at their desired levels after the batch run is finished. $\mathbf{y}(t_f)$ consists of two measured quality variables, concentrations of C_A and C_B, at the end of each batch run (t_f).

The duration of each batch is 150 minutes. The sampling time of each batch is 15 minutes ($K = 10$). Two quality variables in $\mathbf{y}$ are measured at the end of each batch run. The process is affected by a persistent disturbance ($NID(0,1.0^2)$) in the two control valves of nominal upstream pressures. The process measured jacket and the reactor temperatures with noise ($NID(0,0.1^2)$) are included. Also, the initial concentration C_A with noise ($NID(0,0.003^2)$) is introduced. A total of 50 batches based on the normal operation whose design quality is set to be $C_A = 0.344\, lbmol/ft^3$ and $C_B = 0.426\, lbmol/ft^3$ is used as the basis analysis. The quality and the measurement MPLS models are built upon the based case operation data whose designed quality is around the normal condition. Another 20 runs which do not come from the training sets are produced in a similar way for cross validation. Both MPLS models with one principal component can capture the variance in the relationships of the system. To investigate the performance of the proposed method, two cases, including setpoint changes, and disturbance changes, are tested respectively.

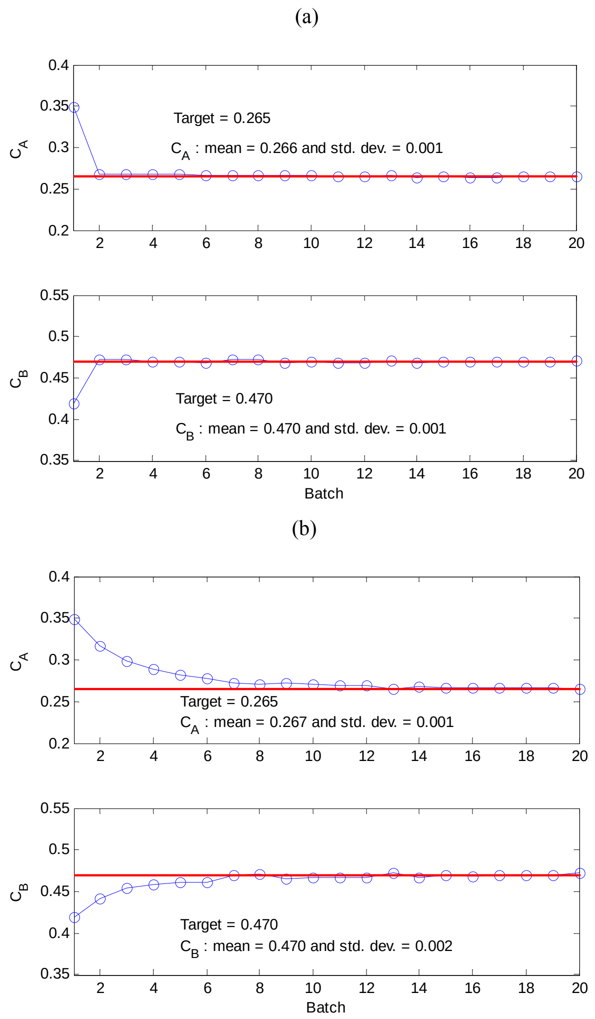


Fig. 4 Convergence of C_A and C_B for the setpoint change using (a) the two-step MPLS based ILC; (b) the standard MPLS based ILC.

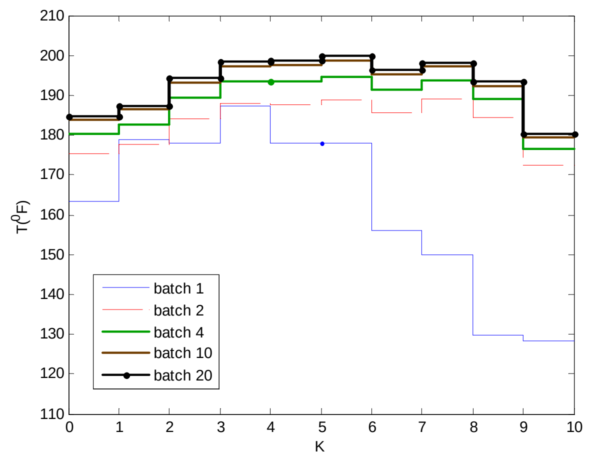
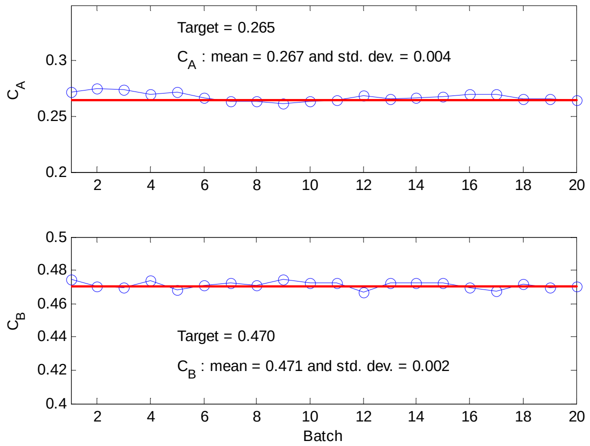


Fig. 5. The evolution of the designed temperature trajectories for the setpoint change using the two-step MPLS based ILC

A. Setpoint Changes

To test the performance of the proposed control strategy, the desired qualities are changed from $C_A = 0.344\, lbmol/ft^3$ and $C_B = 0.426\, lbmol/ft^3$ to $C_A = 0.265\, lbmol/ft^3$ and $C_B = 0.470\, lbmol/ft^3$. The results of the standard MPLS based ILC and the two-step MPLS based ILC strategies are plotted in Fig. 4(a) and (b). Both figures indicate that the control objectives of the two control strategies are improved gradually and they converge to the desired levels. The performance of the two-step MPLS based ILC is corrected within the batch run. Because the standard MPLS model embeds all process variables into a single input data block and a single output data block [6], this will hinder the system's quick response from the required changes at each time point. Thus, the controlled outputs of the two-step MPLS based ILC can approach the new setpoint much faster than those of the standard MPLS based ILC. Fig 5 shows the evolution of the control variable $\mathbf{x}^c$ under the two-step MPLS based ILC control structure that approaches the new setpoint.

(a)



(b)

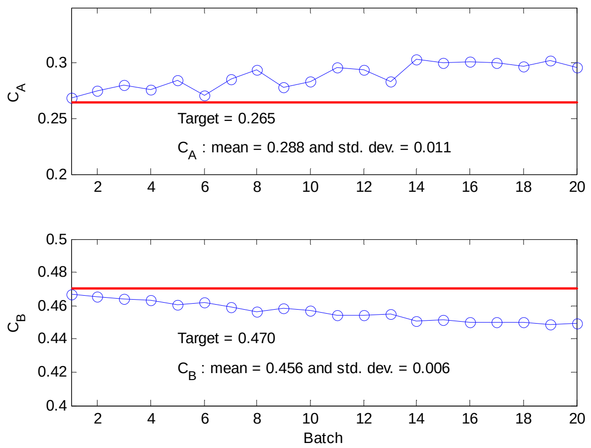


Fig. 5 Convergence of C_A and C_B for the disturbance change: (a) with the update of the two-step MPLS models, (b) without the update of the two-step MPLS models.

B. Disturbance Change

In this case, the rate constants are gradually deactivated after each batch run. The proposed method is used to reject the batch-to-batch deactivated disturbance. Fig. 5(a) shows the results of the control strategy. The desired end-quality is gradually achieved. If the MPLS based ILC control designs without updating the process model are performed, Fig. 5(b) show the rejected disturbance results. Because of the plant-model errors and difficulty in control, the values of the controlled qualities drift away. The controlled final concentrations cannot be converged.

VI. Conclusion

In this paper, two-step MPLS based iterative learning ILC is developed for the tracking control of product quality in the batch operation process. Rather than using detailed knowledge of the operation process model, the proposed strategy depends only on information contained in the historical database of the past batches. Two MPLS models are built. The quality MPLS can be used to calculate the desired on-line measured variables along the time axis. And then using the measurement MPLS, the desired on-line measured variables at each time point can be used to calculate the operating input variables to reach the end point target qualities when the batch is finished. The proposed method, unlike the traditional MIMO control design with the lump structure, explores the benefits of the decomposition MPLS framework in the reduced subspace and the optimal control design of the MIMO system. The MPLS model structure can decompose the MIMO control system into several independent pairs of inputs and outputs. Then, the conventional SISO ILC design strategy can be directly and separately used to determine each SISO system. Further study for improving the control performance, such as the addition of the disturbance estimation and on-line within-batch control for the improvement of the convergence rate, will be conducted in our next research.

References

[1] B. Srinivasan, S. Palanki, and D. Bonvin, "Dynamic optimization of batch processes I. Characterization of the nominal solution," *Comput. Chem. Eng.,* vol. 27, pp. 1-26, Jan. 2003.

[2] R. Luus and O. Rosen, "Application of dynamic programming to final state constrained optimal control problems," *Ind. Eng. Chem. Res.*, vol. 30, pp. 1525-1530, Jul. 1991.

[3] Q. Zhao, B. Yang, Z. Sun, and F. Sun, "Intelligent control scheme of a batch process," in *Proc. of the 3rd World Congress on Intelligent Control and Automation*, Hefei, 2000, pp. 439-443.

[4] N. Shahrokhi and M. R. Pishvaie, "Artificial neural network feedforward/feedback control of a batch polymerization reactor," in *Proc. of the American Control Conference*, Philadelphia, 1998, pp. 3391-3395.

[5] Z. Xiong and J. Zhang, "A batch-to-batch iterative optimal control strategy based on recurrent neural network models," *J. Proc. Cont.,* vol. 15, pp. 11-21, Feb. 2005.

[6] J. Flores-Cerrillo and J. F. MacGregor, "Iterative learning control for final batch product quality using partial least squares models," *Ind. Eng. Chem. Res.,* vol. 44, pp. 9146-9155, Nov. 2005.

[7] J. Chen, and F. Wang, "PLS Based dEWMA Run-to-Run Controller for MIMO Non-Squared Semiconductor Processes," *J. Process Contr.*, vol. 17, pp. 309-319, 2007.

[8] W. L. Luyben,. *Process Modeling, Simulation and Control for Chemical Engineers*. McGraw-Hill, New York, 1990.