Korean J. Chem. Eng., **26**(6), 1512-1518 (2009)
DOI: 10.2478/s11814-009-0322-4

Iterative learning controller synthesis using FIR models for batch processes

Junghui Chen† and Kuo-Tong Chou

R&D Center for Membrane Technology, Department of Chemical Engineering,
Chung-Yuan Christian University, Chung-Li, Taiwan 320, China
(*Received 9 May 2009 • accepted 27 July 2009*)

Abstract−Adaptive iterative learning control based on the measured input-output data is proposed to solve the traditional iterative learning control problem in the batch process. It produces a control law with self-tuning capability by combining a batch-to-batch model estimation procedure with the control design technique. To build the unknown batch operation system, the finite impulse response (FIR) model with the lifted system is constructed for easy construction of a recursive least squares algorithm. It can identify the pattern of the current operation batch. The proposed model reference control method is applied to feedback control of the lifted system. It finds an appropriate control input so that the desired performance of the batch output can track the prescribed finite-time trajectory by iterative trials. Furthermore, on-line tracking control is developed to explore the possible adjustments of the future input trajectories within a batch. This can remove the disturbances in the current batch rather than the next batch trial and keep the product specifications consistent at the end of each batch. To validate the theoretical findings of the proposed strategies, two simulation problems are investigated.

Key words: Batch Process, Iterative Learning Control, Model Reference Control

INTRODUCTION

Iterative learning control (ILC) is one of the most effective control strategies dealing with repetitive tracking control or disturbance rejection in dynamic systems. In general, the ILC system improves its control performance by a self-tuning process without an accurate system model [1-3], because it adjusts the recipe that contains the setpoints of these inputs. It is also believed to be able to produce proper outputs and reduce variability in final products. Phan and Longman [4] used the lifted system description to represent an ILC system as a special type of feedback control loop [4]. Once ILC has been established as a feedback loop, many well-understood feedback design techniques in the past can be applied. However, most of the previous research discussed analysis of learning convergence [5,6]. There has not been much work on the ILC design on the basis of the lifted system representation. The control signals combining system identification and ILC have been refined to enhance the performance of tracking control systems [7,8]. However, the long time-scale development of a detailed mechanistic model might make the model infeasible for agile, responsive manufacturing as the products are typically short-lived and of small-volume [9]. Also, the effectiveness of the real application is likely to be degraded because of the large variation in the operation condition during a batch run. Thus, Balakrishnan and Edgar (2000) used gain-schedule control for a commercial RTP reactor [10]. Huzmezan et al. [11] applied adaptive control to a PVC reactor and an ethoxylated fatty acid reactor [11]. A review paper summarized various optimization strategies of batch processes [12].

With the popularity of computerized measurement devices in the well equipped industrial processes, existing historical data of the measurement profiles are a rich source of information on the variations affecting the process. This indicates that a design technique based on experimental data is a practical way. In this paper, a systematic data-based synthesis of the learning controller on the basis of the lifted system representation is developed for the batch control design. The representation system forms a multivariable feedback structure. The proposed controller produces an iterative learning control law with self-tuning capability by combining the updating procedure of batch-to-batch (BtB) modeling with a control system design technique. The learning strategy of well-known model reference adaptive control is applied. In a traditional model reference control approach, the controller's task makes the system response converge to a reference model. This kind of approach was often used for continuous processes. It has not been much applied to the batch control problem with repetitive operation. Also, in an aging process, the proposed controller is adaptively updated as more data are obtained from repetitive learning. The adaptive iterative learning algorithm is expected to compensate for any slow varying behavior and uncertainties in the batch operation model. In other words, the BtB control strategy is only an open-loop control method over one single batch and the feedback adjustments are applied only between batches rather than during each batch run. This would hinder the system's quick response from eliminating the disturbance at each time point. Therefore, the within-batch (WB) iterative learning control strategy, which allows on-line measurement of variables at the current batch, is developed to explore the possible adjustments of the future input trajectories. It can remove the disturbances without delay and keep the product specifications consistent at the end of each batch.

The remaining paper is structured as follows: The lifted system that represents an ILC system as a feedback control loop between batches is derived in Section 2. Based on this ILC framework, conventional feedback control with proper modification can be applied. Therefore, in Section 3, a controller synthesis technique that formu-

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

lates model reference control is developed. It can track the setpoints and compensate for model uncertainty. In Section 4, on-line WB control is developed to eliminate the disturbances at the current batch, keeping product specifications consistent at the end of each batch. In Section 5, the effectiveness of the proposed method and its potential applications are demonstrated through two computer simulation problems. Finally, concluding remarks are made.

LIFT SYSTEM DESCRIPTION FOR OPERATION BETWEEN BATCHES

A batch process considered in this study can be mathematically formulated as,

$$\mathbf{x}_{k+1}=\mathbf{A}\mathbf{x}_k+\mathbf{b}u_k,\quad 0\le k\le K$$
$$y_k=\mathbf{c}^T\mathbf{x}_k+w_k \tag{1}$$

where **A**, **b** and **c** are parameter matrix and vectors. **x**, u and y are unmeasured state variable, input and output, respectively. w_k is the unmeasured disturbance. K is the duration of each batch run. The mapping the continuous time input (u) to the output (y) over the whole batch run can be represented by a finite impulse response (FIR) model,

$$\mathbf{y}_i=\mathbf{H}\mathbf{u}_i+\mathbf{w}_i \tag{2}$$

where $\mathbf{w}_i$ denotes the unmeasured disturbance sequence. The input ($\mathbf{u}_i$) and the output ($\mathbf{y}_i$) sequences at the interval of K samples are measured,

$$\mathbf{u}_i=[u_i(0)\ u_i(1)\ \cdots\ u_i(K-1)]^T$$
$$\mathbf{y}_i=[y_i(1)\ y_i(2)\ \cdots\ y_i(K)]^T \tag{3}$$

and **H** is a lower triangular matrix whose elements are the Markov parameters of the plant consisting of the impulse response coefficients:

$$\mathbf{H}=\begin{bmatrix} h_1 & 0 & \cdots & \cdots & 0 \\ h_2 & h_1 & & & 0 \\ \vdots & \vdots & \ddots & & \vdots \\ h_{K-1} & h_{K-2} & \cdots & \ddots & 0 \\ h_K & h_{K-1} & \cdots & \cdots & h_1 \end{bmatrix} \tag{4}$$

In a batch system with repetitive operation, the same task is repeated many times and the system returns to the initial operation condition before the next batch run or the trial begins. The batch system is considered as a two-dimensional model. The operation number, denoted by the subscript i, is the execution or operation of the system at the ith trial. The operation time, denoted by the subscript k, is the time evolution of the process behavior during a single batch. For learning control, it is natural to describe how a change in the control input (**u**) from one repetition to the next affects the system response (**y**) from one repetition to the next. The typical form of a linear ILC algorithm is as follows:

$$\mathbf{u}_i=\mathbf{u}_{i-1}+\mathbf{G}_c(\mathbf{G}_f\mathbf{y}^{sp}-\mathbf{y}_i) \tag{5}$$

where $\mathbf{y}^{sp}$ is the desired setpoint vector of a batch run. A two-degree-of-freedom structure with $\mathbf{G}_c$ and $\mathbf{G}_f$ is used to improve the control performance. $\mathbf{G}_c$ and $\mathbf{G}_f$ are the matrices for the feedback and the feedforward controllers,

$$\mathbf{G}_c=\begin{bmatrix} g_0^c & 0 & \cdots & \cdots & \cdots & \cdots & \cdots & 0 \\ g_1^c & g_0^c & 0 & \ddots & \ddots & \ddots & \ddots & \vdots \\ g_2^c & g_1^c & g_0^c & 0 & \ddots & \ddots & \ddots & \vdots \\ \vdots & \ddots & \ddots & \ddots & \ddots & \ddots & \ddots & \vdots \\ g_{L-1}^c & \ddots & \ddots & \ddots & \ddots & \ddots & \ddots & \vdots \\ 0 & g_{L-1}^c & \ddots & \ddots & \ddots & \ddots & \ddots & \vdots \\ \vdots & \ddots & \ddots & \ddots & \ddots & \ddots & \ddots & \vdots \\ 0 & \cdots & 0 & g_{L-1}^c & \cdots & g_2^c & g_1^c & g_0^c \end{bmatrix} \tag{6}$$

$$\mathbf{G}_f=\begin{bmatrix} g_0^f & 0 & \cdots & \cdots & \cdots & \cdots & \cdots & 0 \\ g_1^f & g_0^f & 0 & \ddots & \ddots & \ddots & \ddots & \vdots \\ g_2^f & g_1^f & g_0^f & 0 & \ddots & \ddots & \ddots & \vdots \\ \vdots & \ddots & \ddots & \ddots & \ddots & \ddots & \ddots & \vdots \\ g_{L-1}^f & \ddots & \ddots & \ddots & \ddots & \ddots & \ddots & \vdots \\ 0 & g_{L-1}^f & \ddots & \ddots & \ddots & \ddots & \ddots & \vdots \\ \vdots & \ddots & \ddots & \ddots & \ddots & \ddots & \ddots & \vdots \\ 0 & \cdots & 0 & g_{L-1}^f & \cdots & g_2^f & g_1^f & g_0^f \end{bmatrix} \tag{7}$$

whose elements are from the polynomials of $G_c(q^{-1})$ and $G_f(q^{-1})$ in the shift operator in the time domain, respectively:

$$G_c(q^{-1})=g_0^c+g_1^cq^{-1}+\cdots+g_L^cq^{-L}=\sum_{l=0}^{L}g_l^cq^{-l} \tag{8}$$

$$G_f(q^{-1})=g_0^f+g_1^fq^{-1}+\cdots+g_L^fq^{-L}=\sum_{l=0}^{L}g_l^fq^{-l} \tag{9}$$

After the trial delay operator, z^{-1} is introduced, the previous batch input ($\mathbf{u}_{i-1}$) becomes $\mathbf{u}_{i-1}=z^{-1}\mathbf{u}_i$, where z^{-1}, the discrete delay operator in the batch trial domain, represents the operation with the fixed time point from batch to batch It is contrary to conventional feedback control operating with the fixed batch from time-step to time-step in the time domain. Thus, the ILC control algorithm in Eq. (5) can be represented by,

$$\mathbf{u}_i=\mathbf{M}\mathbf{G}_c(\mathbf{G}_f\mathbf{y}_i^{sp}-\mathbf{H}\mathbf{u}_i) \tag{10}$$

where $\mathbf{M}=\mathrm{diag}\left(\frac{z^{-1}}{1-z^{-1}}\ \cdots\ \frac{z^{-1}}{1-z^{-1}}\right)$ is a memory block which can retain values of the past batch runs. Fig. 1 shows the relationship between the inputs and the outputs of the closed-loop system from one batch run to the next one. It consists of the model of the open-loop plant, the feedback and the feedforward controllers. The control structure is similar to the conventional feedback control loop. It differs in the forms of the data. Information from each batch run is grouped into an augmented vector which will represent the measurement at a single time point in conventional feedback control. In Fig. 1, the system is a mapping from an input vector to an output

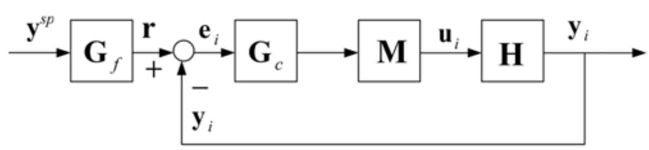

Fig. 1. ILC closed loop structure used for the trial invariant trajectory.

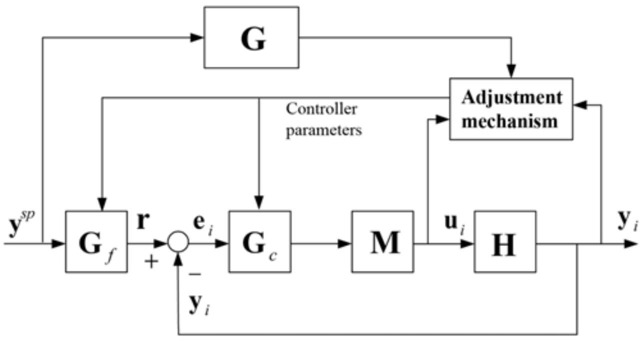

Fig. 2. ILC with a model reference.

vector. The error vector is the difference between the current batch output vector and the desired batch reference. Then this information is used to make an adjustment of the input vector. The memory loop can store the input vector of the previous trials for input calculation of the next trial and reduce the error in the next run. The equation of the closed-loop batch-to-batch system becomes

$$\mathbf{y}_i=[\mathbf{I}+\mathbf{HMG}_c]^{-1}\mathbf{HMG}_c\mathbf{r} \tag{11}$$

BATCH-TO-BATCH ITERATIVE LEARNING CONTROL

The learning control objective is to determine the controllers, $\mathbf{G}_c$ and $\mathbf{G}_f$, and generate appropriate control input data that can produce a detailed output history through iterative trials. A reference model is used as a prescribed model to guide the iterative learning process shown in Fig. 2,

$$\mathbf{y}_i=\mathbf{G}\mathbf{y}^{sp} \tag{12}$$

where $\mathbf{G}$ is any desired response model that describes a causal system. Thus, the matrix is a lower triangular Toeplitz matrix; each diagonal contains the same value and all values above the main diagonal are zero. Let the feedback controller represent two parts ($\mathbf{W}$ and $\mathbf{S}$),

$$\mathbf{G}_c=\mathbf{W}^{-1}\mathbf{S} \tag{13}$$

To match the format of $\mathbf{G}_c$, $\mathbf{W}$ and $\mathbf{S}$ are defined as,

$$\mathbf{W}=\begin{bmatrix} w_0 & 0 & \cdots & \cdots & \cdots & 0 \\ w_1 & w_0 & 0 & \ddots & \cdots & \vdots \\ \vdots & \vdots & \ddots & \ddots & \ddots & \vdots \\ w_{L-1} & w_{L-2} & w_{L-3} & \ddots & \ddots & \vdots \\ 0 & w_{L-1} & w_{L-2} & \cdots & w_0 & 0 \\ 0 & 0 & w_{L-1} & w_{L-2} & \cdots & w_0 \end{bmatrix} \tag{14}$$

$$\mathbf{S}=\begin{bmatrix} s_0 & 0 & \cdots & \cdots & \cdots & 0 \\ s_1 & s_0 & 0 & \ddots & \cdots & \vdots \\ \vdots & \vdots & \ddots & \ddots & \ddots & \vdots \\ s_{L-1} & s_{L-2} & s_{L-3} & \ddots & \ddots & \vdots \\ 0 & s_{L-1} & s_{L-2} & \cdots & s_0 & 0 \\ 0 & 0 & s_{L-1} & s_{L-2} & \cdots & s_0 \end{bmatrix} \tag{15}$$

Eq. (11) becomes,

$$\mathbf{y}_i=[\mathbf{W}+\mathbf{SMH}]^{-1}\mathbf{SMHr} \tag{16}$$

The controller matrices, $\mathbf{W}$ and $\mathbf{S}$, can be specified directly for the desired performance (Eq. (12)),

$$\mathbf{G}^{-1}=\mathbf{W}+\mathbf{SMH} \tag{17}$$

where the inverse of $\mathbf{G}$ is given by,

$$\mathbf{G}^{-1}\equiv\begin{bmatrix} d_1 & 0 & \cdots & \cdots & 0 \\ d_2 & d_1 & & & 0 \\ \vdots & \vdots & \ddots & & \vdots \\ d_{K-1} & d_{K-2} & \cdots & \ddots & 0 \\ d_K & d_{K-1} & \cdots & \cdots & d_1 \end{bmatrix} \tag{18}$$

As a result, the equation of the closed-loop system becomes,

$$\mathbf{y}_i=\mathbf{GCr} \tag{19}$$

where $\mathbf{C}=\mathbf{G}^{-1}-\mathbf{W}$. Consequently, the design goal is complete with a choice of

$$\mathbf{G}_f=\mathbf{C}^{-1} \tag{20}$$

With the given desired matrix $\mathbf{G}$, and a set of input-output data $\mathbf{H}$, an appropriate pair, $\mathbf{G}_c$ and $\mathbf{G}_f$, is computed by using Eqs. (13) and (20) so that Eq. (12) can be satisfied.

By packing $\bar{\mathbf{S}}=\mathbf{SM}$,

$$\bar{\mathbf{S}}=\begin{bmatrix} \bar{s}_0 & 0 & \cdots & \cdots & \cdots & 0 \\ \bar{s}_1 & \bar{s}_0 & 0 & \ddots & \cdots & \vdots \\ \vdots & \vdots & \ddots & \ddots & \ddots & \vdots \\ \bar{s}_{L-1} & \bar{s}_{L-2} & \bar{s}_{L-3} & \ddots & \ddots & \vdots \\ 0 & \bar{s}_{L-1} & \bar{s}_{L-2} & \cdots & \bar{s}_0 & 0 \\ 0 & 0 & \bar{s}_{L-1} & \bar{s}_{L-2} & \cdots & \bar{s}_0 \end{bmatrix} \tag{21}$$

and $\bar{s}_j=s_j z^{-1}(1-z^{-1})^{-1}$, j=1, 2, $\cdots$, L. An alternative form of Eq. (17) can be expressed as,

$$\mathbf{Ax}=\mathbf{d} \tag{22}$$

with

$$\mathbf{x}=[\bar{s}_0\ \bar{s}_0\ \cdots\ \bar{s}_{L-1}\ w_0\ w_1\ \cdots\ w_{L-1}]^T \tag{23}$$

$$\mathbf{A}=\begin{bmatrix} h_1 & 0 & \cdots & 0 & 1 & 0 & \cdots & 0 \\ h_2 & h_1 & \ddots & \vdots & 0 & 1 & \ddots & \vdots \\ \vdots & \ddots & \ddots & 0 & \vdots & \ddots & \ddots & 0 \\ h_L & \cdots & h_2 & h_1 & 0 & \cdots & 0 & 1 \\ h_{L+1} & \cdots & h_3 & h_2 & 0 & 0 & \cdots & 0 \\ h_{L+2} & \cdots & h_4 & h_3 & 0 & \ddots & \ddots & \vdots \\ \vdots & \cdots & \vdots & \vdots & \vdots & \ddots & \ddots & 0 \\ h_K & \cdots & h_{K-L+1} & h_{K-L} & 0 & \cdots & 0 & 0 \end{bmatrix} \tag{24}$$

$$\mathbf{d}=[d_1\ d_2\ \cdots\ d_K]^T \tag{25}$$

With the given $\mathbf{d}$ and $\mathbf{A}$, an estimation formula can be obtained,

$$\mathbf{x}=[\mathbf{AA}^T]^{-1}\mathbf{A}^T\mathbf{d} \tag{26}$$

The hat notation used as the prediction has been removed for simplicity. After $\mathbf{x}=[\bar{s}_0\ \bar{s}_1\ \cdots\ \bar{s}_{L-1}\ w_0\ w_1\ \cdots\ w_{L-1}]^T$ is substituted into Eqs. (13) and (20), the controllers $\mathbf{G}_c$ and $\mathbf{G}_f$ can be formulated. The above control algorithm is well suited for the learning control application when the batch process can be reasonably described in advance. However, the proposed approach should be able to adapt the controllers to accommodate changes in the process. Fig. 2 represents the structure of a self-tuning regulator, which constitutes a way of adjusting the controllers. It is composed of two loops. The inner loop acts as the original ILC controller for the system with changing behavior during each trial. In the outer loop, the parameters of the ILC controller are adjusted to perform the recursive model estimator mechanism.

At Batch i, the system of Eq. (2) can be conveniently arranged into another matrix form,

$$\mathbf{y}_i=\mathbf{U}_i\mathbf{h}_i+\mathbf{w}_i \tag{27}$$

where

$$\mathbf{U}_i=\begin{bmatrix} u_i(0) & 0 & \cdots & \cdots & 0 \\ u_i(1) & u_i(0) & 0 & \cdots & 0 \\ \vdots & \vdots & \ddots & \ddots & \vdots \\ u_i(K-2) & u_i(K-3) & \cdots & \ddots & 0 \\ u_i(K-1) & u_i(K-2) & \cdots & u_i(1) & u_i(0) \end{bmatrix} \tag{28}$$

$$\mathbf{h}_i=[h_1\ h_2\ \cdots\ h_K]^T \tag{29}$$

With all the available batch data, the impulse coefficients $\mathbf{h}_i$ can be computed

$$\mathbf{h}_i=\mathbf{F}_i^{-1}\mathbf{U}^T\mathbf{y} \tag{30}$$

where

$$\mathbf{U}=\begin{bmatrix} \mathbf{U}_1 \\ \mathbf{U}_2 \\ \vdots \\ \mathbf{U}_i \end{bmatrix},\ \mathbf{F}_i=\mathbf{U}^T\mathbf{U},\ \mathbf{y}=\begin{bmatrix} \mathbf{y}_1 \\ \mathbf{y}_2 \\ \vdots \\ \mathbf{y}_i \end{bmatrix} \tag{31}$$

Suppose that the batch data from Batch i+1 ($\mathbf{y}_{i+1}$ and $\mathbf{U}_{i+1}$) become available. The recursive least-squares estimator can be derived as follows:

$$\mathbf{h}_{i+1}=\mathbf{h}_i+\mathbf{F}_{i+1}\mathbf{U}_{i+1}^T(\mathbf{y}_{i+1}-\mathbf{U}_{i+1}\mathbf{h}_i) \tag{32}$$

$$\mathbf{F}_{i+1}=\frac{1}{\lambda}(\mathbf{F}_i-\mathbf{F}_i\mathbf{U}_{i+1}^T(\lambda\mathbf{I}+\mathbf{U}_{i+1}\mathbf{F}_i\mathbf{U}_{i+1}^T)^{-1}\mathbf{U}_{i+1}\mathbf{F}_i) \tag{33}$$

where λ is the forgetting factor that places heavier emphasis on more recent data when the operation system is batch-varying.

With the updated model parameters ($\mathbf{h}_{i+1}$), the recursive solution for calculating the new controller ($\mathbf{x}_{i+1}$) can also be derived,

$$\mathbf{x}_{i+1}=\mathbf{x}_i+\Psi_{i+1}\mathbf{A}_{i+1}^T\mathbf{e}_{i+1} \tag{34}$$

where $\Psi_{i+1}=\frac{1}{\lambda}(\Psi_i-\Psi_i\mathbf{A}_i^T(\lambda\mathbf{I}+\mathbf{A}_i\Psi_i\mathbf{A}_i^T)^{-1}\mathbf{A}_{i+1}\Psi_i)$ and $\Psi_1=[\mathbf{A}_1\mathbf{A}_1^T]^{-1}$. $\mathbf{e}_{i+1}=\mathbf{d}-\mathbf{A}_{i+1}\mathbf{x}_i$.

WITHIN-BATCH ON-LINE CONTROL

If the measurement variables at each time point can be measured during each batch run, the within-batch on-line control strategy can be implemented and improvement can be made by adjustment at each sampling time point. Suppose only measurements of the output and the input variables from the beginning of the batch till the current time (k) are available, then the measurement function in the matrix form can be written,

$$\mathbf{y}(k)=\mathbf{U}(k)\mathbf{h}(k)+\mathbf{w}(k) \tag{35}$$

where

$$\mathbf{U}(k)=\begin{bmatrix} \begin{bmatrix} u_{i-1}(0) & 0 & \cdots & \cdots & 0 \\ u_{i-1}(1) & u_{i-1}(0) & 0 & \cdots & 0 \\ \vdots & \vdots & \ddots & \ddots & \vdots \\ u_{i-1}(K-2) & u_{i-1}(K-3) & \cdots & \ddots & 0 \\ u_{i-1}(K-1) & u_{i-1}(K-2) & \cdots & u_{i-1}(1) & u_{i-1}(0) \end{bmatrix} \\ \begin{bmatrix} u_i(0) & 0 & \cdots & \cdots & 0 \\ u_i(1) & u_i(0) & 0 & \cdots & 0 \\ \vdots & \vdots & \ddots & \ddots & \vdots \\ u_i(K-2) & u_i(K-3) & \cdots & \ddots & 0 \\ u_i(K-1) & u_i(K-2) & \cdots & u_i(1) & u_i(0) \end{bmatrix} \end{bmatrix}_{(K+k)\times(K+k)} \tag{36}$$

$$\mathbf{h}(k)=[h_1\ h_2\ \cdots\ h_k]^T \tag{37}$$

$$\mathbf{y}(k)=[[y_{i-1}(1)\ y_{i-1}(2)\ \cdots\ y_{i-1}(K)][y_i(1)\ y_i(2)\ \cdots\ y_i(k)]]^T \tag{38}$$

Here the data of the previous batch run is provided to enhance the robustness of the parameters and get the best fit by making use of more measurements. The derivation of this recursive formula is straightforward. Readers are referred to adaptive control and system identification literature [13,14]. Thus, only the results are given here without proof,

$$\mathbf{h}(k+1)=\mathbf{h}(k)+\mathbf{F}^{-1}(k+1)\mathbf{u}(k+1)(y(k+1)-\mathbf{u}(k+1)^T\mathbf{h}(k)) \tag{39}$$

$$\mathbf{F}^{-1}(k+1)=\mathbf{F}^{-1}(k)-\frac{\mathbf{F}^{-1}(k)\mathbf{u}(k+1)\mathbf{u}(k+1)^T\mathbf{F}^{-1}(k)}{1+\mathbf{u}(k+1)^T\mathbf{F}^{-1}(k)\mathbf{u}(k+1)} \tag{40}$$

where $\mathbf{F}(k)=\mathbf{U}^T(k)\mathbf{U}(k)$ and $\mathbf{u}(k+1)^T=[u(k)\ u(k-1)\ \cdots\ u(1)\ u(0)]$.

According to Eq. (26), once $\mathbf{d}$ is specified, the following equation in the matrix form must be solved in order to compute $\mathbf{W}$ and $\mathbf{S}$ at the time point k,

$$\mathbf{A}(k)\mathbf{x}(k)=\mathbf{d}(k) \tag{41}$$

where

$$\mathbf{x}(k)=[\bar{s}_0\ \bar{s}_0\ \cdots\ \bar{s}_{L-1}\ w_0\ w_1\ \cdots\ w_{L-1}]^T \tag{42}$$

$$\mathbf{A}(k)=\begin{bmatrix} \begin{bmatrix} h_1 & 0 & \cdots & 0 & 1 & 0 & \cdots & 0 \\ h_2 & h_1 & \ddots & \vdots & 0 & 1 & \ddots & \vdots \\ \vdots & \ddots & \ddots & 0 & \vdots & \ddots & \ddots & 0 \\ h_L & \cdots & h_2 & h_1 & 0 & \cdots & 0 & 1 \\ h_{L+1} & \cdots & h_3 & h_2 & 0 & 0 & \cdots & 0 \\ h_{L+2} & \cdots & h_4 & h_3 & 0 & \ddots & \ddots & \vdots \\ \vdots & \cdots & \vdots & \vdots & \vdots & \ddots & \ddots & 0 \\ h_K & \cdots & h_{K-L+1} & h_{K-L} & 0 & \cdots & 0 & 0 \end{bmatrix}_{i-1} \\ \begin{bmatrix} h_1 & 0 & \cdots & 0 & 1 & 0 & \cdots & 0 \\ h_2 & h_1 & \ddots & \vdots & 0 & 1 & \ddots & \vdots \\ \vdots & \ddots & \ddots & 0 & \vdots & \ddots & \ddots & 0 \\ h_L & \cdots & h_2 & h_1 & 0 & \cdots & 0 & 1 \\ h_{L+1} & \cdots & h_3 & h_2 & 0 & 0 & \cdots & 0 \\ h_{L+2} & \cdots & h_4 & h_3 & 0 & \ddots & \ddots & \vdots \\ \vdots & \cdots & \vdots & \vdots & \vdots & \ddots & \ddots & 0 \\ h_K & \cdots & h_{K-L+1} & h_{K-L} & 0 & \cdots & 0 & 0 \end{bmatrix}_{i} \end{bmatrix} \tag{43}$$

$$\mathbf{d}(k)=[[d_1\ d_2\ \cdots\ d_{K-1}]_{i-1}[d_2\ d_2\ \cdots\ d_{k-1}]_i]^T \tag{44}$$

Thus, the recursive computation of the controller parameters in real time applications can also be derived,

$$\mathbf{x}(k+1)=\mathbf{x}(k)+\mathbf{P}^{-1}(k+1)\mathbf{a}(k+1)(d(k+1)-\mathbf{a}(k+1)^T\mathbf{x}(k)) \tag{45}$$

$$\mathbf{P}^{-1}(k+1)=\mathbf{P}^{-1}(k)-\frac{\mathbf{P}^{-1}(k)\mathbf{a}(k+1)\mathbf{a}(k+1)^T\mathbf{P}^{-1}(k)}{1+\mathbf{a}(k+1)^T\mathbf{P}^{-1}(k)\mathbf{a}(k+1)} \tag{46}$$

where $\mathbf{P}(k)=\mathbf{A}(k)^T\mathbf{A}(k)$ and

$$\mathbf{a}(k+1)^T=\begin{cases}\left[\underbrace{h_k\ \cdots\ h_1}_{k}\underbrace{0\ \ldots\ 0}_{L-1}1\underbrace{0\ \ldots\ 0}_{L-k}\right] & k\le L\\ \left[\underbrace{h_k\ h_{k-1}\ \cdots\ h_{k-L}}_{L}\underbrace{0\ 0\ \cdots\ 0}_{L}\right] & k>L\end{cases} \tag{47}$$

With the updated $\mathbf{x}(k+1)$, the corresponding $\mathbf{G}_f$ and $\mathbf{G}_c$ are calculated to implement to the system at the next step. The whole procedure is repeated until the batch is completed. Then BtB control is applied to calculating the input in the new batch run before the next phase of WB control begins.

SIMULATION ILLUSTRATION

In this section, two simulation studies, including a simulation plant with a math model and a batch reactor without any prior knowledge of the process models, are conducted to demonstrate the performance of the proposed learning control method for controller synthesis.

1. Example 1: Math Model Plant

In this example, a plant is represented by a third-order process with the transfer function ($\tilde{G}_p(s)$),

$$\tilde{G}_p(s)=\frac{1.8}{(s+1)(s+2)(s+9.5)} \tag{48}$$

The reference trajectory used in this example is,

$$y^{sp}(k)=\sin(0.24k)+\text{square}(0.3k),\ k=0, 1, 2, \ldots, 99 \tag{49}$$

where *square* represents a square wave with a period of 2π. The reference model for the control design is,

$$G_m(s)=\frac{1}{(\tau s+1)^2} \tag{50}$$

The control goal is to achieve stable and fast reference tracking without oscillations. Thus, the time constant (τ) in the reference model cannot be very low to avoid the oscillation response, but if it is too conservative, the controlled response would be slow. In this example, $\tau=0.8$ is selected to meet our requirement. The nominal model always suffers uncertainty. It is assumed that the plant changes to,

$$G_p(s)=\frac{1.8+\Delta}{(s+1)(s+2)(s+9.5)} \tag{51}$$

where the gain error $\Delta=1.0$. When the model is not updated, a set of the learning control actions which is designed based on Eqs. (13) and (20) is implemented on the linear plant. In Fig. 3, the output of the first batch is still far away from the design setpoint. After several batch runs, the controlled outputs are plotted densely near the thick line. They are getting closer and closer to the design setpoint.

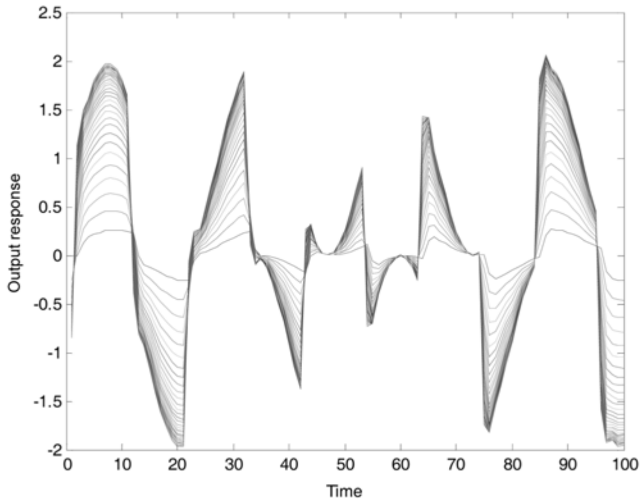

Fig. 3. Performance of each batch controlled outputs for tracking the setpoint marked in a thick line when the changed model is not updated. Each thin line represents the control output of its batch run.

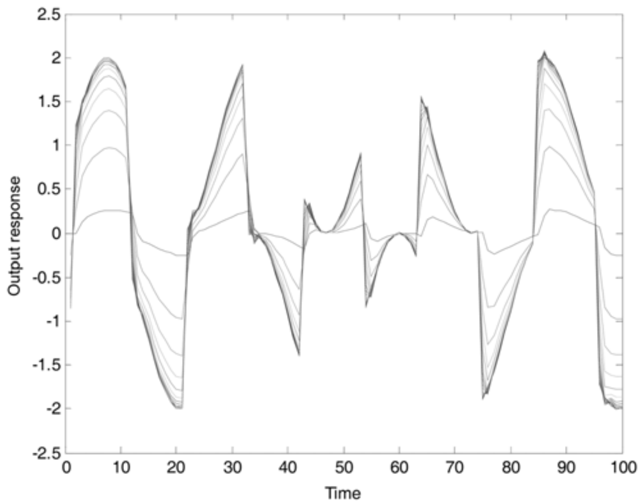

Fig. 4. Performance of each batch controlled outputs for tracking the setpoint marked in a thick line when the BtB updating model is applied. Each thin line represents the control output of its batch run.

With the available operation batch data, the model for the operating plant is updated. Figs. 4 and 5 show the evolution of the controlled outputs for the control strategies based on the two updating techniques. They indicate that the updated model of BtB control can improve the control performance. BtB control is expected to correct the poor performance in the next batch runs. The controlled output approaches the setpoint after ten batch trials. Furthermore, instead of using the fixed control variables, the updated model of WB control can quickly change the designed control variables after four batch runs and track the reference setpoint.

2. Example 2: Batch Reactor

A reactant is fed into the batch reactor. Assume it is blended well in the reactor and the liquid density is kept constant during the opera-

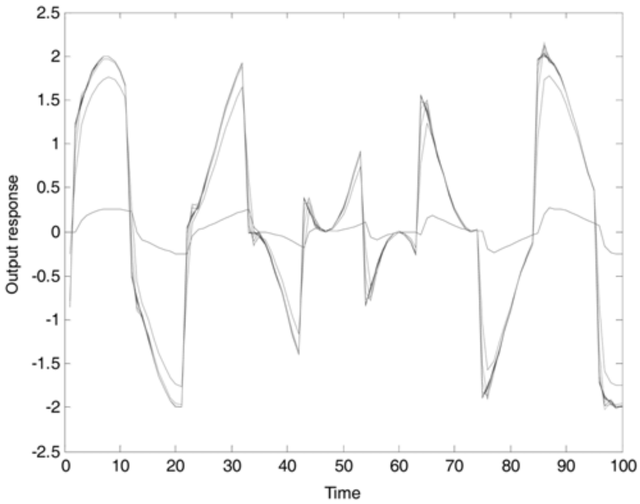

Fig. 5. Performance of each batch controlled output for tracking the setpoint marked in a thick line when WB updating model is applied. Each thin line represents the control output of its batch run.

tion. A consecutive reaction takes place in the reactor:

$$A \xrightarrow{k_1} B \xrightarrow{k_2} C \tag{52}$$

where A→B has the second-order kinetics and B→C has the first-order kinetics. The dynamics of the batch reactor can be referred to [15]. In this example, the desired temperature trajectory is,

$$T_d(t)=54+71\exp(-2.5\times10^{-3}t) \tag{59}$$

In this case, temperature control of the nonlinear chemical batch reactor is considered. Before the proposed iterative learning control method is applied, the finite impulse response model can be estimated from the past operation batch runs. Based on the initial model parameters, the iterative learning controller can be computed with Eqs. (13) and (20), where the optional reference model is specified to achieve the required response speed,

$$G_m(s)=\frac{1}{(0.5s+1)^2} \tag{60}$$

To test the performance for such a control law, the heat transfer term (U_j) is changed to,

$$U_j(i)=1.16e^{\frac{-i}{100}} \tag{61}$$

where the heat transfer would be gradually decayed after each batch run (i). The effectiveness of the proposed learning control method is clearly illustrated in Fig. 6, in which the tracking error is the mean squared error (MSE) between the controlled and the reference outputs for each batch trial, $MSE_i=\frac{1}{N}\sqrt{\sum_{k=1}^{N}(y^{sp}(k)-y_i(k))^2}$. At the 30th iteration, perfect tracking is achieved. Of course, the performance can be improved when BtB and WB control with the updated models are applied. By observing the responses of the control performance, faster response can be achieved especially when WB control is conducted. Thus, ILC based on the on-line updated model is crucial to improvement of performance.

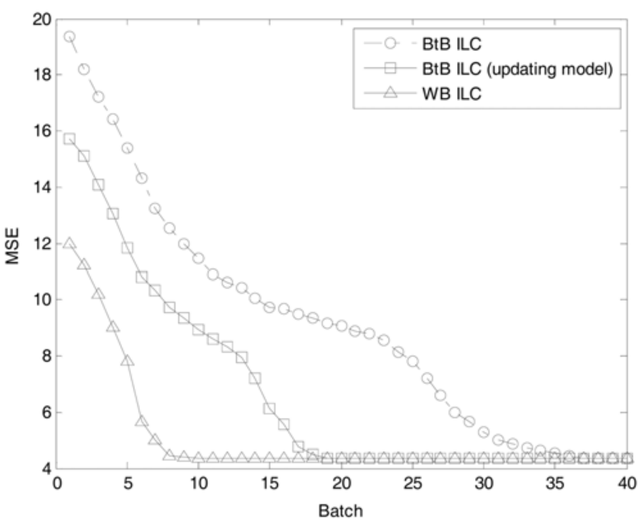

Fig. 6. Comparison of the proposed ILC convergence histories for the three control strategies in Example 2: fixed model (line with circle), BtB updating model (line with square), WB updating model (line with triangle).

CONCLUSIONS

ILC is operated along a certain trajectory during one trial to improve the performance when the system is operated along the same trajectory at the next trial. A data-based controller synthesis of ILC batch systems has been presented. The major advantages of the proposed method are

(1) Based on the dynamics of ILC from trial to trial, ILC can be systematically and naturally represented by a feedback system in the trial domain. This setting allows simple and easy design of the batch system in the trial domain based on the rich and well-understood feedback techniques, resulting in good performance with minimal effort.

(2) The model reference adaptive technique is developed for ILC of batch processes. It can be regarded as an adaptive servo system whose desired batch performance is expressed in terms of a reference model. It gives the desired response to a batch setpoint output.

(3) Because the convergence performance of ILC is sensitive to model errors, the model quality strongly influences the behavior of ILC. The proposed ILC combining the updated model and the model reference adaptive technique is considered. The desired performance of ILC can be achieved even though the controlled unit is not known a priori.

(4) To quickly reject or reduce the disturbances in the revolving batch rather than the end of the batch, on-line control is also developed.

In this work, an ILC based on the FIR model has been presented to serve as a basic premise for the system. Several issues warrant future research to expand the generalized feedback setting in the trial domain, including the extended theory to be used for a time variant system with multivariables and the enhanced computation for the big impulse response matrices when there is a very long trajectory.

ACKNOWLEDGMENT

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-98-CR-CE, and the National Science Council (NSC) for their financial support.

REFERENCES

1. D. S. Boning, W. P. Moyne, T. H. Smith, J. Moyne, R. Telfeyan, A. Hurwitz, S. Shellamn and J. Taylor, *IEEE Transactions on Components, Packaging, and Manufacturing Technology*, Part C, **19**, 307 (1996).
2. K. S. Lee, I. S. Chin, H. J. Lee and J. H. Lee, *AIChE J.*, **45**, 2175 (1999).
3. M. Mezghani, G. Roux, M. Cabassud, M. V. Le Lann, B. Dahhou and G. Casamatta, *IEEE Transactions on Control Systems Technology*, **10**, 822 (2002).
4. M. A. Phan and R. W. Longman, *A mathematical theory of learning control for linear discrete multivariable systems*, Proceedings of the AIAA/AAS Astrodynamics Conference, AIAA Publishers, Minneapolis, MN (1988).
5. K. L. Moore, *Multi-loop control approach to designing iteratively learning controllers*, Proceedings of the 37th IEEE Conference on Decision and Control, Tampa, Florida (1998).
6. N. Amann, D. H. Owens and E. Rogers, *International Journal of Control*, **65**, 277 (1996).
7. W. H. Cho, T. F. Edgar and J. T. Lee, *Korean J. Chem. Eng.*, **26**, 307 (2009).
8. K. Han, K. J. Park, H. Chae and E. S. Yoon, *Korean J. Chem. Eng.*, **24**, 921 (2007).
9. Z. Xiong and J. Zhang, *Journal of Process Control*, **15**, 11 (2005).
10. K. S. Balakrishnan and T. F. Edgar, *Journal of Thin Solid Films*, **365**, 322 (2000).
11. M. Huzmezan, B. Gough and S. Kovac, *Advanced control of batch reactor temperature*, American Control Conference (2002).
12. D. Bonvin, B. Srinivasan and D. Ruppen, *AIChE Symposium Series*, **326**, 255 (2002).
13. G. C. Goodwin and K. S. Sin, *Adaptive filtering prediction and control*, Prentice-Hall Inc., Englewood Cliffs, NJ (1984).
14. L. Ljung, *System identification: Theory for the user*, Prentice-Hall Inc., Englewood Cliffs, NJ (1987).
15. W. H. Ray and S. Szelely, *Process optimization*, Wiley, New York (1973).