2005 American Control Conference
June 8-10, 2005. Portland, OR, USA

ThA06.3

An Integrated Batch-to-batch Iterative Learning Control and within Batch Control Strategy for Batch Processes

Zhihua Xiong[1], Jie Zhang[2], Xiong Wang[1], Yongmao Xu[1]

[1] Department of Automation, Tsinghua University, Beijing 100084, P. R. China
Email: zhxiong@tsinghua.edu.cn
[2] School of Chemical Engineering and Advanced Materials, University of Newcastle, Newcastle upon Tyne, NE1 7RU, U. K., E-mail: jie.zhang@ncl.ac.uk

Abstract— **An integrated batch-to-batch iterative learning control (ILC) and within batch on-line shrinking horizon model predictive control (SHMPC) strategy for the tracking control of product qualities in batch processes is proposed. ILC is used for batch-to-batch control based on a batch-wise linear time-varying (LTV) perturbation model and the convergence of batch-wise tracking error under ILC is guaranteed. On-line SHMPC within a batch can reduce the effects of disturbances immediately and improve the performance of the current batch run. The on-line model prediction can be also obtained based on the batch-wise LTV model. The integrated control strategy can complement both methods to obtain good performance of tracking control. The proposed strategy is illustrated on a simulated batch polymerization process.**

I. INTRODUCTION

BATCH-to-batch control exploits the repetitive nature of batch processes to refine the operating policy. Recently, iterative learning control (ILC) has been used to directly update input trajectory [1]. ILC can update the control trajectory for the next batch run using the information from previous batch runs so that the output trajectory converges asymptotically to the desired reference trajectory [2]-[5].

However, batch-to-batch control can only improve the performance of future batch runs and cannot improve the performance of the current batch run. Batch-wise control also cannot handle disturbances that change from batch to batch in a completely random fashion and may actually amplify their effect [6]. On the other hand, if output variables can be measured or inferred accurately on-line, it is possible to implement on-line control that adjusts the control policy for the remaining batch period while the batch is going on [7]. Shrinking horizon model predictive control (SHMPC) [8] is most suitable for on-line control of batch processes within the current batch. Because on-line SHMPC can respond to disturbances immediately and batch-to-batch ILC can correct any bias left uncorrected by the on-line controller, it is natural to explore the possibility of combining both methods to obtain good control performance. The integrated control strategy can complement each other to render the benefits of both methods [7],[9]. If disturbances occur, the integrated control method is expected to diminish more rapidly the effect of disturbances than results for only implementing ILC from batch to batch.

Based on a batch-wise linear time-varying (LTV) perturbation model, the ILC method can be implemented from batch to batch for tracking trajectories and the convergence of tracking error is guaranteed [5]. On-line SHMPC control within a batch can be established in a manner similar to the batch-to-batch control formulation. The batch-wise LTV perturbation model can be also utilized, and predictive errors of the immediate previous batch run are utilized to modify predictions of the predictive model during the current batch run.

The rest of this paper is organized as follows: Section 2 presents batch-to-batch ILC based on a batch-wise LTV perturbation model. Section 3 presents a method to obtain future model predictions within a batch based on the LTV model and then on-line SHMPC method is implemented. The integrated control strategy of combining both strategies is outlined in Section 4. Application of integrated control strategy to a simulated batch polymerization process is given in Section 5. Finally Section 6 draws some concluding remarks.

II. BATCH-TO-BATCH CONTROL

We consider batch processes where the batch run length (t_f) is fixed and consists of N sampling intervals and all batches run from the same initial condition. The batch-to-batch control problem is to manipulate the whole control profile so that the product quality variables follow specific desired reference trajectories. It would be convenient to consider a batch-wise static function relating the control profile to the product quality profile over the whole batch duration. It can be written in matrix form as

$$\mathbf{Y}_k = \mathbf{F}(y_0,\mathbf{U}_k) + \mathbf{v}_k \quad (1)$$

where the subscript k denotes the batch index, $\mathbf{Y}_k=[y_k^T(1), y_k^T(2),\ldots,y_k^T(N)]^T$ ($y\in R^n$, $n\geq1$), is a matrix of product quality variables and can be obtained on-line or off-line, y_0 is the initial value, $\mathbf{U}_k=[u_k(0), u_k(1),\ldots,u_k(N-1)]^T$ ($u\in R^m$, m=1 in this work) is the manipulated variable, $F(\cdot)$ represents the non-linear static functions, and $\mathbf{v}_k$ is a matrix of measurement noises. Subtracting the time-varying nominal trajectories from the process operation trajectories removes the majority of the process non-linearity and allows linear modeling methods to perform well on the resulting perturbation variables [8].

2.1 Batch-wise LTV Perturbation Model

Linearizing the non-linear batch process model described by Eq(1) with respect to control sequence around the nominal trajectories, the following can be obtained

$$\mathbf{Y}_k = \mathbf{Y}_s + \left.\frac{\partial \mathbf{F}(y_0,\mathbf{U}_k)}{\partial \mathbf{U}_k}\right|_{\mathbf{U}_s}(\mathbf{U}_k - \mathbf{U}_s) + \mathbf{w}_k + \mathbf{v}_k \quad (2)$$

where $\mathbf{U}_s$ is the nominal control trajectory, $\mathbf{Y}_s$ is the nominal product quality trajectory and $y_s(0) = y_0$, and $\mathbf{w}_k$ is a sequence of model errors due to the linearisation (i.e., due to neglecting the higher oder terms). Then a batch- wise LTV *perturbation* model can be obtained as

$$\overline{\mathbf{Y}}_k = \mathbf{G}_s \overline{\mathbf{U}}_k + \mathbf{d}_k \quad (3)$$

where $\mathbf{G}_s=(\partial\mathbf{F}(y_0,\mathbf{U})/\partial\mathbf{U})\,|_{(\mathbf{U}_k=\mathbf{U}_s)}$, $\overline{\mathbf{U}}_k=\mathbf{U}_k-\mathbf{U}_s$ and $\overline{\mathbf{Y}}_k = \mathbf{Y}_k-\mathbf{Y}_s$ are *perturbation variables* of control and product quality variables and $\overline{y}_k(0)$=0, and $\mathbf{d}_k=\mathbf{w}_k+\mathbf{v}_k$ is the model disturbance sequence, respectively. $\mathbf{G}_s$ is batch-wise linear time-varying in the sense that it varies with $\mathbf{U}_s$, which usually varies from batch to batch. Due to the causality (i.e. the product quality variables at time t is a function of only all control actions up to time t), the structure of $\mathbf{G}_s$ is restricted to the following lower-block-triangular form:

$$\mathbf{G}_s = \begin{bmatrix} g_{10} & 0 & \cdots & 0 \\ g_{20} & g_{21} & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ g_{N0} & g_{N1} & \cdots & g_{NN-1} \end{bmatrix} \quad (4)$$

where $g_{ij}\in R^n$.

The batchwise LTV model $\mathbf{G}_s$ can be found by linearizing a non-linear model along the nominal trajectories or through direct identification from process operational data. Available methods for identifying $\mathbf{G}_s$ range from simple static linear regression, such as the least squares and its variants (e.g., partial least squares, PLS) [5],[10], to more elaborate optimal dynamic estimation methods like the Kalman filtering [6].

2.2 Batch-to-batch Iterative Learning Control

In this study, we utilize the model errors of the immediate previous batch run to modify predictions of the perturbation model. The *model prediction* and *modified prediction of perturbation model* in the $(k+1)^{\text{th}}$ batch run is obtained as

$$\hat{\overline{\mathbf{Y}}}_{k+1}^{ILC} = \hat{\mathbf{G}}_s \overline{\mathbf{U}}_{k+1} \quad (5)$$

$$\tilde{\overline{\mathbf{Y}}}_{k+1}^{ILC} = \hat{\overline{\mathbf{Y}}}_{k+1}^{ILC} + \hat{\boldsymbol{\varepsilon}}_k^{ILC} \quad (6)$$

where $\hat{\boldsymbol{\varepsilon}}_k^{ILC}=\overline{\mathbf{Y}}_k-\hat{\overline{\mathbf{Y}}}_k^{ILC}$ and superscript ILC represents batch-to-batch *iterative learning control*. Considering that the objective of ILC is to track the desired reference trajectories of product quality, the *tracking errors of process* and *modified perturbation model prediction* are defined respectively as

$$\mathbf{e}_k^{ILC} = \overline{\mathbf{Y}}_d - \overline{\mathbf{Y}}_k^{ILC} \quad (7)$$

$$\tilde{\mathbf{e}}_k^{ILC} = \overline{\mathbf{Y}}_d - \tilde{\overline{\mathbf{Y}}}_k^{ILC} \quad (8)$$

where $\overline{\mathbf{Y}}_d=\mathbf{Y}_d-\mathbf{Y}_s$, and $\mathbf{Y}_d$ is the specified reference trajectory and assumed here to be set reasonably. Then an iterative relationship for $\tilde{\mathbf{e}}_k^{ILC}$ along the batch index k can be obtained as [5]

$$\tilde{\mathbf{e}}_{k+1}^{ILC} = \mathbf{e}_{k+1}^{ILC} - \hat{\mathbf{G}}_s\Delta\overline{\mathbf{U}}_{k+1}^{ILC} \quad (9)$$

where $\Delta\overline{\mathbf{U}}_{k+1}^{ILC}=\overline{\mathbf{U}}_{k+1}^{ILC}-\overline{\mathbf{U}}_k^{ILC}$ represents the input change between two adjacent batch runs. Given the above batch-wise error transition model, the objective of ILC is to design a learning algorithm to manipulate the control policy so that the product qualities follow the specific desired reference trajectories from batch to batch. We consider solving the following quadratic objective function based on the modified prediction errors upon the completion of the k^{th} batch run to update the input trajectory for the $(k+1)^{\text{th}}$ batch run

$$\min_{\Delta\overline{\mathbf{U}}_{k+1}^{ILC}} J_{k+1}^{ILC} = \frac{1}{2}[\tilde{\mathbf{e}}_{k+1}^{ILC^T}\mathbf{Q}_s\tilde{\mathbf{e}}_{k+1}^{ILC} + \Delta\overline{\mathbf{U}}_{k+1}^{ILC^T}\mathbf{R}_s\Delta\overline{\mathbf{U}}_{k+1}^{ILC}] \quad (10)$$

where $\mathbf{Q}_s$ and $\mathbf{R}_s$ are positive definitive matrices and selected here as $\mathbf{Q}_s=\lambda_q\cdot\mathbf{I}_N$ and $\mathbf{R}_s=\lambda_r\cdot\mathbf{I}_N$.

Through straightforward manipulation, the following ILC law can be obtained

$$\overline{\mathbf{U}}_{k+1}^{ILC} = \overline{\mathbf{U}}_k^{ILC} + \hat{\mathbf{K}}^{ILC}\mathbf{e}_k^{ILC} \quad (11)$$

where $\hat{\mathbf{K}}^{ILC} = [\hat{\mathbf{G}}_s^T\mathbf{Q}_s\hat{\mathbf{G}}_s + \mathbf{R}_s]^{-1}\hat{\mathbf{G}}_s^T\mathbf{Q}_s$.

The convergence of ILC alogrithm can be obtained and its proof can directly be derived from the convergence theorems in literature. In this study, we obtain that $\mathbf{e}_k^{ILC}$ will nominally converge as $k\to\infty$ if $\mathbf{I}-\hat{\mathbf{G}}_s\hat{\mathbf{K}}^{ILC}$ has all its eigenvalues inside the unit circle [5], i.e. $\|\mathbf{I}-\hat{\mathbf{G}}_s\hat{\mathbf{K}}^{ILC}\|<1$.

III. Online Control Within Batch

Batch-to-batch ILC strategy can only improve the performance of future batch runs and cannot improve the performance of current batch run. In addition, it cannot handle disturbances that change from batch to batch in a completely random fashion and may actually amplify their effect [6]. However, if on-line measurement of output variables can be made on a reliable basis, one can explore the possibility of implementing on-line control that adjusts the future input policy while the batch is going on [7]. On-line batch control can be established in a manner similar to the batch- to-batch control formulation. SHMPC can be utilized within the current batch [8]. In SHMPC, the horizon of model prediction p is equal to the control horizon m and the both decrease while time t passes within the current batch, i.e. $m=p=N-t$.

During on-line SHMPC, it is quite useful to update the future control profile based on the calculated batch-to- batch ILC profile $\overline{\mathbf{U}}_{k+1}^{ILC}$, instead of directly calculating the future control action. It can be represented by

$$\overline{\mathbf{U}}_{k+1}^{OLC}(t+m)=\delta\overline{\mathbf{U}}_{k+1}^{OLC}(t+m)+\overline{\mathbf{U}}_{k+1}^{ILC}(t+m) \qquad (12)$$

where superscript *OLC* represents *on-line control*, $m=N-t$, $\overline{\mathbf{U}}_{k+1}^{OLC}(t+m)=[\overline{u}_{k+1}^{OLC}(t),\ldots,\overline{u}_{k+1}^{OLC}(t+m\text{-}1)]^T$ is a vector of future m control actions to be calculated, $\overline{\mathbf{U}}_{k+1}^{ILC}(t+m)=[\overline{u}_{k+1}^{ILC}(t), \overline{u}_{k+1}^{ILC}(t+1),\ldots,\overline{u}_{k+1}^{ILC}(t+m\text{-}1)]^T$ is a vector of control values in the same m horizons that have been calculated by ILC, and $\delta\overline{\mathbf{U}}_{k+1}^{OLC}$ is the deviation.

3.1 Within Batch Model Prediction

During the on-line control within the current batch run, the future values $\overline{y}_{k+1}^{OLC}(t+m)$ from time t+1 to the end point N must be predicted by using future input sequence $\overline{\mathbf{U}}_{k+1}^{OLC}(t+m)$. The batch-wise LTV perturbation model can be also utilized. Considering the causality in Eq(3) and Eq(4) and due to $\overline{y}_k(0)=0$, the model prediction of the product quality variables at time j (j=1,2,…,N) is a function of all control actions up to time t, i.e. $\hat{\overline{y}}_k(j)=g_j^T\overline{\mathbf{U}}_k(j)$, where g_j is the jth row vector of $\mathbf{G}_s$ in Eq(4), $\overline{\mathbf{U}}_k(j)=[\overline{u}_k(0),\ldots,\overline{u}_k(j-1)]^T$. Therefore at time t in the on-line control, if the control profile is partitioned according to time t, then the model prediction of future time $t+i$ (i=1,…,m) can be written as

$$\hat{\overline{y}}_{k+1}^{OLC}(t+i\,|\,t)=g_{t+i}^1\overline{\mathbf{U}}_{k+1}^{OLC}(t)+g_{t+i}^2\overline{\mathbf{U}}_{k+1}^{OLC}(t+i) \qquad (13)$$

where $g_{t+i}^1=[g_{t+i,0}, g_{t+i,1},\ldots, g_{t+i,t}]^T$, $g_{t+i}^2=[g_{t+i,t+1}, g_{t+i,t+2},\ldots, g_{t+i,t+i}]^T$, $\overline{\mathbf{U}}_{k+1}^{OLC}(t)=[\overline{u}_{k+1}^{OLC}(0),\ldots,\overline{u}_{k+1}^{OLC}(t\text{-}1)]^T$.

$\mathbf{G}_s$ is partitioned according to time t as a block matrix, and then Eq(13) can be written in the matrix form as

$$\begin{bmatrix}\overline{\mathbf{Y}}_{k+1}^{OLC}(t)\\ \overline{\mathbf{Y}}_{k+1}^{OLC}(t+m)\end{bmatrix}=\begin{bmatrix}\hat{\mathbf{G}}_t^{11} & \hat{\mathbf{G}}_t^{12}\\ \hat{\mathbf{G}}_t^{21} & \hat{\mathbf{G}}_t^{22}\end{bmatrix}\begin{bmatrix}\overline{\mathbf{U}}_{k+1}^{OLC}(t)\\ \overline{\mathbf{U}}_{k+1}^{OLC}(t+m)\end{bmatrix} \qquad (14)$$

where $\overline{\mathbf{Y}}_{k+1}^{OLC}(t)$ and $\overline{\mathbf{U}}_{k+1}^{OLC}(t)$ are obtained and calculated before time t, and

$$\hat{\mathbf{G}}_t^{11}=\begin{bmatrix}g_{1,0} & 0 & \cdots & 0\\ g_{2,0} & g_{2,1} & \cdots & 0\\ \vdots & \vdots & \ddots & \vdots\\ g_{t,0} & g_{t,1} & \cdots & g_{t,t-1}\end{bmatrix},\ \hat{\mathbf{G}}_t^{12}=0,$$

$$\hat{\mathbf{G}}_t^{21}=\begin{bmatrix}g_{t+1,0} & g_{t+1,1} & \cdots & g_{t+1,t-1}\\ g_{t+2,0} & g_{t+2,1} & \cdots & g_{t+2,t-1}\\ \vdots & \vdots & \ddots & \vdots\\ g_{t+m,0} & g_{t+m,1} & \cdots & g_{t+m,t-1}\end{bmatrix}, \qquad (15)$$

$$\hat{\mathbf{G}}_t^{22}=\begin{bmatrix}g_{t+1,t} & 0 & \cdots & 0\\ g_{t+2,t} & g_{t+2,t} & \cdots & 0\\ \vdots & \vdots & \ddots & \vdots\\ g_{t+m,t} & g_{t+m,t} & \cdots & g_{t+m,t+m-1}\end{bmatrix}$$

Then we have

$$\hat{\overline{\mathbf{Y}}}_{k+1}^{OLC}(t+m\,|\,t)=\hat{\mathbf{G}}_t^{21}\overline{\mathbf{U}}_{k+1}^{OLC}(t)+\hat{\mathbf{G}}_t^{22}\overline{\mathbf{U}}_{k+1}^{OLC}(t+m) \qquad (16)$$

where $\hat{\overline{\mathbf{Y}}}_{k+1}^{OLC}(t+m\,|\,t)=[\hat{\overline{y}}_{k+1}^{OLC}(t+1|t),\ldots,\hat{\overline{y}}_{k+1}^{OLC}(t+m|t)]^T$.

3.2 On-line Shrinking Horizon MPC

If the predictive errors calculated from the batch-to- batch controller are not added to the predictive model within a batch, on-line control calculation ends up ‘undoing’ the correction made by the batch-to-batch controller [6]. To improve the accuracy of the predictive model, predictive errors of the immediate previous batch run are utilized to modify predictions of the predictive model in the current batch run, which is defined as

$$\tilde{\overline{\mathbf{Y}}}_{k+1}^{OLC}(t+m|t)=\hat{\overline{\mathbf{Y}}}_{k+1}^{OLC}(t+m\,|\,t)+\hat{\boldsymbol{\varepsilon}}_k^{OLC}(t+m|t) \qquad (17)$$

where $\hat{\boldsymbol{\varepsilon}}_k^{OLC}(t+m|t)=\overline{\mathbf{Y}}_k(t+m)-\hat{\overline{\mathbf{Y}}}_k^{OLC}(t+m\,|\,t)$.

Substitute Eq(12) and Eq(16) to Eq(17), then $\tilde{\overline{\mathbf{Y}}}_{k+1}^{OLC}(t+m|t)$ can be rewritten further as

$$\begin{aligned}\tilde{\overline{\mathbf{Y}}}_{k+1}^{OLC}(t+m|t)=\ &\hat{\mathbf{G}}_t^{21}\overline{\mathbf{U}}_{k+1}^{OLC}(t)+\hat{\mathbf{G}}_t^{22}\overline{\mathbf{U}}_{k+1}^{ILC}(t+m)\\ &+\hat{\mathbf{G}}_t^{22}\delta\overline{\mathbf{U}}_{k+1}^{OLC}(t+m)+\hat{\boldsymbol{\varepsilon}}_k^{OLC}(t+m|t)\end{aligned} \qquad (18)$$

The *tracking error* of the modified predictive model for the remaining input moves is defined as

$$\tilde{\mathbf{e}}_k^{OLC}(t+m|t) = \overline{\mathbf{Y}}_d(t+m) - \tilde{\overline{\mathbf{Y}}}_{k+1}^{OLC}(t+m|t) \quad (19)$$

where $\overline{\mathbf{Y}}_d(t+m)=[\bar{y}_d(t+1),...,\bar{y}_d(t+m)]^T$.

The objective function of the SHMPC is defined as

$$\min_{\delta\overline{\mathbf{U}}_{k+1}^{OLC}(t+m)} J_{k+1}^{OLC}(t) = \frac{1}{2}[\tilde{\mathbf{e}}_{k+1}^{OLC^T}(t+m|t)\mathbf{Q}_t\tilde{\mathbf{e}}_{k+1}^{OLC}(t+m|t) \quad (20)$$
$$+\delta\overline{\mathbf{U}}_{k+1}^{OLC^T}(t+m)\mathbf{R}_t\delta\overline{\mathbf{U}}_{k+1}^{OLC}(t+m)]$$

where $\mathbf{Q}_t$ and $\mathbf{R}_t$ are positive definitive weighting matrices with appropriate dimensions. Through straightforward manipulation, the following on-line SHMPC law within batch can be obtained

$$\delta\overline{\mathbf{U}}_{k+1}^{OLC}(t+m) = \hat{\mathbf{K}}_t^{OLC}\boldsymbol{\eta}_{k+1}(t+m) \quad (21)$$

where $\boldsymbol{\eta}_{k+1}(t+m) = \overline{\mathbf{Y}}_d(t+m) - \hat{\mathbf{G}}_t^{21}\overline{\mathbf{U}}_{k+1}^{OLC}(t) - \hat{\mathbf{G}}_t^{22}\overline{\mathbf{U}}_{k+1}^{ILC}(t+m) - \hat{\boldsymbol{\varepsilon}}_k^{OLC}(t+m|t)$, $\hat{\mathbf{K}}_t^{OLC} = [\hat{\mathbf{G}}_t^{22^T}\mathbf{Q}_t\hat{\mathbf{G}}_t^{22} + \mathbf{R}_t]^{-1}\hat{\mathbf{G}}_t^{22^T}\mathbf{Q}_t$.

Then according to Eq(12), the following SHMPC law can be obtained

$$\overline{\mathbf{U}}_{k+1}^{OLC}(t+m) = \overline{\mathbf{U}}_{k+1}^{ILC}(t+m) + \hat{\mathbf{K}}_t^{OLC}\boldsymbol{\eta}_{k+1}(t+m) \quad (22)$$

Only the first element of $\overline{\mathbf{U}}_{k+1}^{OLC}(t+m)$ is applied to the process and the same procedure is repeated with t increased by 1 but control horizon m shrinked by 1. This SHMPC law is similar to that in Lee *et al*. [6] except for the term $\boldsymbol{\eta}_{k+1}(t+m)$ instead of $\mathbf{e}_k(t|t)$.

IV. Integrated Control

Because on-line SHMPC can respond to disturbances immediately and batch-to-batch ILC can correct any bias left uncorrected by the on-line controller, it is natural to explore the possibility of combining them to obtain good performance of tracking trajectories. The integrated control strategy can complement each other to render the benefits of the both. The procedure of integrated control by combing batch-to-batch ILC with on-line SHMPC within batch is outlined as follows:

Step 1. Based on the historical process operation data set, select the nominal input and output trajectories ($\mathbf{U}_s$,$\mathbf{Y}_s$). Initially set k=0 and $\overline{\mathbf{U}}_k^{ILC} = \overline{\mathbf{U}}_k^{OLC} = \mathbf{U}_s$.

Step 2. Based on $\overline{\mathbf{U}}_k^{ILC}$, use batch-to-batch ILC method to calculate the whole control trajectory $\overline{\mathbf{U}}_{k+1}^{ILC}$ of the (k+1)th batch run. The model prediction errors $\hat{\varepsilon}_k^{ILC}$ are calculated and used to correct the batch-wise model predictions. Then based on the modified predictions $\tilde{\overline{\mathbf{Y}}}_{k+1}^{ILC}$, a new control policy $\overline{\mathbf{U}}_{k+1}^{ILC}$ for the next batch is calculated by using the ILC law Eq(11).

Step 3. During the (k+1)th batch, at time t (t=1,…,N), based on errors $\hat{\boldsymbol{\varepsilon}}_k^{OLC}(t+m|t)$ calculated in the previous batch and the calculated $\overline{\mathbf{U}}_{k+1}^{ILC}(t+m)$ by ILC, the future control policy $\overline{\mathbf{U}}_{k+1}^{OLC}(t+m)$ is obtained by using the SHMPC method Eq(22). Then its first element is applied to the process. Repeat this procedure in the current batch run until time t reaches the end of batch. After the (k+1)th batch run, the output profile $\overline{\mathbf{Y}}_{k+1}$ and the whole on-line control policy $\overline{\mathbf{U}}_{k+1}^{OLC}$ are obtained.

Step 4. Set $\overline{\mathbf{U}}_{k+1}^{ILC} = \overline{\mathbf{U}}_{k+1}^{OLC}$ and k=k+1, return to step 2.

It has been shown that convergence holds under some reasonable assumption in batch-to-batch ILC [5]. SHMPC can further improve the performance of batch control within the batch. Robustness of integrated control has not been investigated rigorously [6]. However, experience from extensive numerical studies demonstrates that integrated control has good performance.

V. Application To A Simulated Batch Polymerization Reactor

This example involves a thermally initiated bulk polymerization of styrene in a batch reactor. The differential equations describing the polymerization process are given by Kwon and Evans [11] through reaction mechanism analysis and laboratory testing. Gattu and Zafiriou [12] report the parameter values of the first principle model. The product quality variables include the conversion (y_1), the dimensionless number-average chain lengths (NACL, y_2) and dimensionless weight-average chain lengths (WACL, y_3). The input variable is $u=T/T_{ref}$, where T is the absolute temperature of the reactor and T_{ref} is the reference value. In this study, the final time t_f is fixed to be 313 minutes, and initial values of the outputs are $y_1(0)$=0, $y_2(0)$=1, and $y_3(0)$ =1. The desired product reference trajectory $\mathbf{Y}_d$ was taken from [12].

Here the batch length is divided into N equal stages and two values of N are studied, N = 10 and N = 5. Thirteen batches of process operation under different temperature profiles were simulated from the mechanistic model and used as the historical process data sets for building relationship between u and $\boldsymbol{y}=[y_1, y_2, y_3]^T$. A batch-wise LTV perturbation model $\mathbf{G}_s$ is utilized to build the relationship between u and $\boldsymbol{y}$.

To investigate the performance of the proposed integrated control strategy, it is compared with the simple batch-to-batch ILC scheme. The parameters in ILC were set as $\mathbf{Q}=\mathbf{I}_N$ and $\mathbf{R}=0.05\mathbf{I}_N$, while the parameters in integrated control were set as $\mathbf{Q}_t=\mathbf{I}_m$ and $\mathbf{R}_t=0.15\mathbf{I}_m$, where $m \leq N$ and it is shrinking as time t increases within a batch. The results under two control strategies are shown in Fig. 1 and Fig. 2.

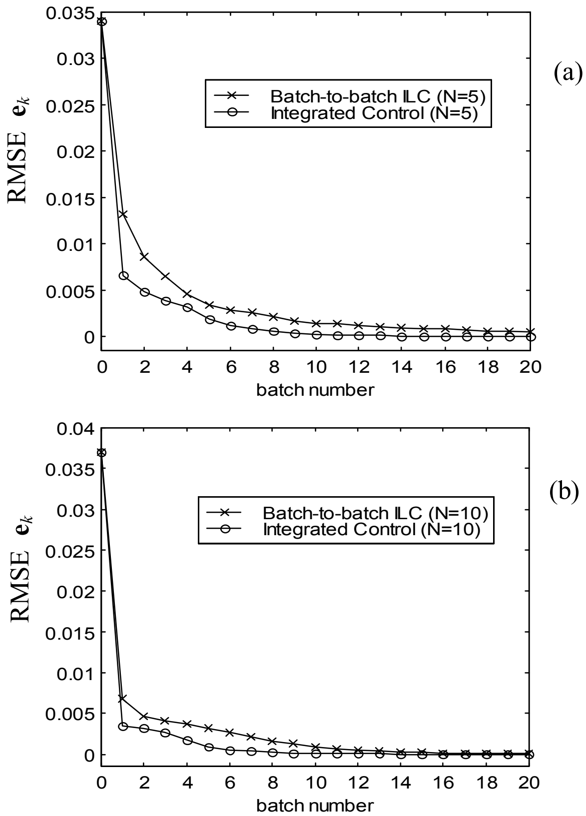

Fig. 1. Convergence of RMSE under two strategies in different time stages: (a) N=5 (b) N=10

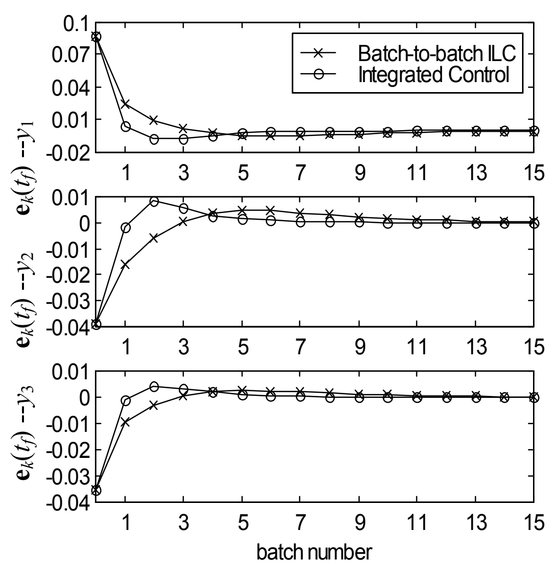

Fig. 2. Convergence of $\mathbf{e}_k(t_f)$ under two strategies (N=10): (a) y_1 (b) y_2 (c) y_3

Although the parameters to be estimated when N=10 are more than when N=5, the model is more accurate than when N=5 and the results for N=10 are slightly better than those for N=5 under two strategies. Fig. 1 shows the RMSE of tracking error of product quality $\mathbf{e}_k$ under two strategies at different time stages. Since the final product quality is of the main interest in batch process operation, the tracking errors $\mathbf{e}_k(t_f)$ at the batch end-point from these two strategies are also compared, as shown in Fig. 2. It shows that $\mathbf{e}_k(t_f)$ is also improved gradually while the whole trajectory converges asymptotically to the desired trajectory.

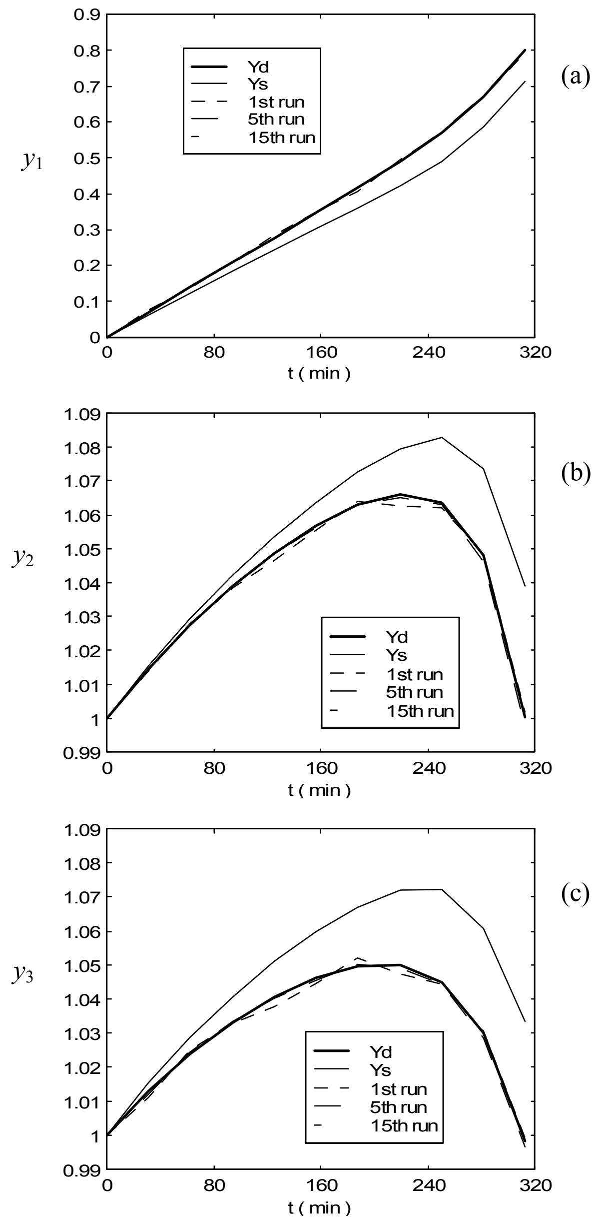

Fig. 3. Trajectories of quality variables under integrated control (N=10): (a) y_1; (b) y_2 ; (c) y_3

It can also be seen that both the RMSE of $\mathbf{e}_k$ and $\mathbf{e}_k(t_f)$ have almost converged after about 5 batch runs in integrated control strategy, but they converge after 11 batch runs in batch-to-batch ILC scheme. Fig. 3 and Fig. 4 show, respectively, the product quality profiles $\mathbf{Y}_k$ and control profile $\mathbf{U}_k$ of the 1st, 5th and 15th batch runs when N=10. Fig. 5 shows that during the 1st batch run under the integrated control, $\overline{\mathbf{U}}_1^{ILC}$ was calculated by the ILC law before the batch run began, but the control action at each time stage was slightly updated by SHMPC and then implemented to the process, finally the control profile was changed to $\overline{\mathbf{U}}_1^{OLC}$. From the results, it can be seen that $\mathbf{Y}_k$ has converged to the desired reference $\mathbf{Y}_d$ after 5 batch runs under the integrated control strategy. The performance of tracking product qualities is also improved under the integrated control strategy than under the simple batch-to-batch ILC. As can be seen, under the integrated control scheme, its advantage of error correction within batch is combined with the benefit of gradual reduction to the minimum error afforded by the batch-to-batch control.

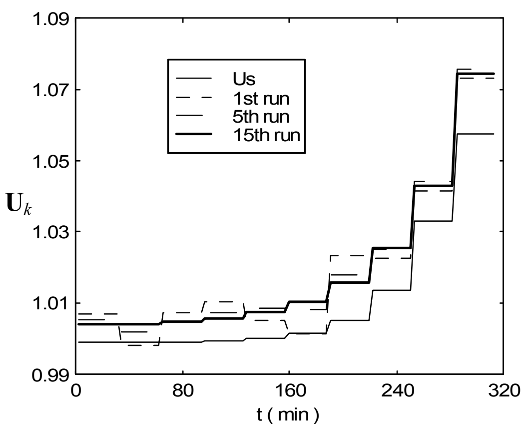

Fig. 4. Convergence of $\mathbf{U}_k$ under integrated control

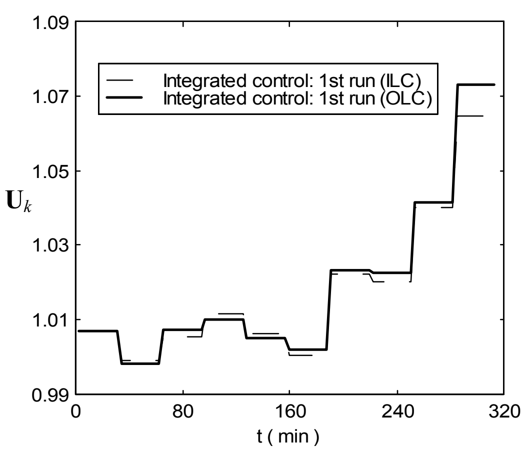

Fig. 5. Comparison of $\overline{\mathbf{U}}_{k+1}^{ILC}$ and $\overline{\mathbf{U}}_{k+1}^{OLC}$ in the 1st batch run under integrated control strategy

VI. Conclusions

An integrated control strategy for the tracking control of product quality in batch processes is proposed by combing batch-to-batch iterative learning control with on-line shrinking horizon model predictive control. On-line SHMPC within batch can decrease the effects of model-plant mismatch, while batch-to-batch ILC can correct the bias left uncorrected by the on-line controller. The integrated control strategy can complement each other to obtain good performance of tracking trajectories. The proposed strategy is illustrated on a simulated batch polymerization process. The results have demonstrated that the performance of tracking product qualities can be improved quite well under the integrated control strategy than under the simple batch-to-batch ILC.

Acknowledgements

The authors greatly appreciate the financial support of the NSFC (No. 60404012).

References

[1] N. Amann, D.H. Owens and E. Rogers , "Predictive optimal iterative learning control," *Int. J. Control*, vol. 69, pp 203-226, 1998.

[2] I. S. Chin, K.S. Lee, and J.H. Lee, "A technique for integrated quality control, profile control, and constraint handling for batch processes," *Ind. Eng. Chem. Res.*, vol. 39, pp 693-705, 2000

[3] J. H. Lee, K. S. Lee and W.C. Kim, "Model- based iterative learning control with a quadratic criterion for time-varying linear systems," *Automatica*, vol. 36, pp 641-657, 2000.

[4] F. J. Doyle, C.A. Harrison, and T.J. Crowley, "Hybrid model-based approach to batch-to-batch control of particle size distribution in emulsion polymerization," *Computers Chem. Engng.*, vol. 27, pp 1153-1163, 2003.

[5] Z. H. Xiong, and J. Zhang, "Product quality trajectory tracking of batch processes using iterative learning control based on time- varying perturbation model," *Ind. Eng. Chem. Res.*, vol. 42, pp 6802-6814, 2003.

[6] K. Lee, J. H. Lee, D. R. Yang *et al*. "Integrated run-to-run and on-line model-based control of particle size distribution for a semi-batch precipitation reactor," *Computers Chem. Engng.*, vol. 26, pp 1117-1131, 2002.

[7] K. S. Lee and J.H. Lee, "Iterative learning control-based batch process control technique for integrated control of end product properties and transient profiles of process variables," *Journal of Process Control*, vol. 13, pp 607–621, 2003.

[8] S. A. Russell, P. Kesavan and J.H. Lee, "Recursive data-based prediction and control of batch product quality," *AIChE J.*, vol. 44, pp 2442-2458, 1998.

[9] J. Flores-Cerrillo and J.F. MacGregor, "Within batch and batch-to-batch inferential adaptive control of semibatch reactors: a partial least squares approach," *Ind. Eng. Chem. Res.*, vol. 42, pp 3334-3345, 2003.

[10] B. Srinivasan, D. Bonvin, E. Visser *et al.*, "Dynamic optimization of batch processes II: role of measurements in handling uncertainty," *Computers Chem. Engng.*, vol. 27, pp 27-44, 2003.

[11] Y. D. Kwon and L.B. Evans, "A coordinate transformation method for the numerical solution of nonlinear minimum-time control problems," *AIChE J.*, vol. 21, pp 1158-1164,1975

[12] G. Gattu and E. Zafiriou, "A methodology for on-line setpoint modification for batch reactor control in the presence of modeling error," *Chemical Engineering Journal*, vol. 75, pp 21-29, 1999.