ADCHEM 2006

International Symposium on Advanced Control of Chemical Processes
Gramado, Brazil – April 2-5, 2006

2D MODEL PREDICTIVE ITERATIVE LEARNING CONTROL SCHEMES FOR BATCH PROCESSES

Jia Shi[1,2], **Furong Gao**[1], **Tie-Jun Wu**[2]

[1]*Deptment of Chemical Engineering, Hong Kong University of Science & Technology, Hong Kong, China*
[2]*Institute of Intelligent Systems and Decision Making, Zhejiang University, Hangzhou, 310027, China*

Abstract: Iterative learning control (ILC) system is modelled and treated as a 2D system in this paper. Based on single-batch and multi-batch cost functions, 2D model predictive iterative learning control (2D-MPILC) schemes are developed in the framework of model predictive control (MPC) for the 2D system. Structure analysis shows that the resulted 2D-MPILC laws are causal and they implicitly combine a time-wise MPC law with a cycle-wise ILC law to ensure the optimal control in 2D sense. To eliminate oscillating input, 2D control penalty is introduced to the 2D-MPILC design. The simulation results show that the proposed schemes are effective.

Keywords: Iterative methods, Learning control, Two-dimensional (2D) system, Model-based control, Predictive control, Batch control.

1. INTRODUCTION

Iterative learning control (ILC) was originally developed for the manipulation of an industrial robot to repetitively perform a given task (Arimoto *et al.*, 1984). Since then, it has been widely studied and extended to applications on processes with repetitive or cyclic nature (Xu *et al.*, 1998).

The conventional ILC scheme (Arimoto *et al.*, 1984), which uses only the information of previous cycles for control input refinement, is only a batch-to-batch control that may not be able to guarantee the control performance along time index. For this reason, real-time feedback control is often combined with the conventional ILC to ensure the control performance not only along the time but also along the cycle. In the early works using such a combination (Xu *et al.*, 1998), however, the separate designs of real-time feedback control and cycle-wise ILC were performed; this may not be able to guarantee optimal performance for both directions. Norm-optimal ILC that implicitly combines a state feedback control with a feed-forward ILC was proposed by Amann *et al.* (1995, 1996) based on the performance index defined over one cycle. Their extended method includes prediction over future cycles (Owens *et al.*, 2000), the resulted feedback control law also in state form, however, is non-causal, and the computation load may be heavy for long batch duration. Base on a quadratic performance defined over one cycle, Lee *et al.* (2000) proposed a model-based ILC scheme.

*Corresponding author: Telephone: +852-2358 7139.
Fax: +852-2358 0054. Email: kefgao@ust.hk.

An ILC system essentially is a two-dimensional (2D) system, where dynamic behaviour along the time is determined by the process and the feedback control, while ILC introduces dynamics along the cycle. Model and design ILC system as a 2D system can result in a united design of time-wise feedback control and cycle-wise ILC, guaranteeing the control performance in 2D sense. The above reviewed methods did not treat the optimizations of ILC design directly in 2D sense. While the existing 2D based methods (Geng *et al.*, 1990; Kurek *et al.*, 1993; Shi *et al.*, 2005) consider only the convergence and/or robustness of the system, leading to, sometime, a conservative control law.

In this paper, the iterative learning control design is treated from the 2D system viewpoint. Single-batch and multi-batch objective functions are defined and optimized in the framework of model predictive control (MPC) for the 2D system, resulting in a single-batch and a multi-batch 2D model predictive iterative learning control (2D-MPILC) schemes. Structure analysis is conducted to give insight of the resulted 2D control system. It shows that the resulted 2D-MPILC schemes consist of two types of controls: one is an MPC that uses the real-time input-output information to ensure the control performance within cycle, and the other is batch-wise ILC that improves the control performance from cycle to cycle. The united design of these two types of controls ensures the optimal control in terms of the defined 2D cost functions. The computation of the methods depends on the design parameters that can be balanced by the practitioner, and the resulted control laws are casual for practical implementation. 2D input change penalty terms can also be easily introduced to the design to ensure a smooth control. Simulation results show that the proposed methods are very effective.

2. PROBLEM FORMULATIONS

For simplicity, it is assumed in this paper that the underlying process is a SISO system. All the results can be extended to MIMO cases.

2.1 Batch processes and ILC law

A batch process, repetitively performing a given task over finite time duration, called a batch or cycle, is described by the following linear model

$$\Sigma_{BP}: \quad A(q^{-1})y_k(t) = B(q^{-1})u_k(t) + w_k(t) \tag{1}$$
$$t = 0,1,...,T; \quad k = 1,2,...$$

where t is time and k represents the batch/cycle index, $u_k(t)$, $y_k(t)$ and $w_k(t)$ are, respectively, the input, output and disturbance of the process at time t in the k th cycle, q^{-1} indicates the *unit backward-shift operator*, and $A(q^{-1})$ and $B(q^{-1})$ are both operator polynomials

$$A(q^{-1}) = 1 + a_1q^{-1} + a_2q^{-2} + \cdots + a_nq^{-n} \tag{2}$$

$$B(q^{-1}) = b_1q^{-1} + b_2q^{-2} + \cdots + b_mq^{-m} \tag{3}$$

For the above batch process, introduce an ILC law with the form

$$u_k(t) = u_{k-1}(t) + r_k(t), \qquad u_0(t) = 0, \quad t = 1,2,...,T \tag{4}$$

where $r_k(t)$ is referred as the *updating law* to be determined, and $u_0(t)$ is the *initial profile* of iteration.

2.2 2D representation and cost functions

Substituting (4) into (1) leads to the following input-output model

$$\Sigma_{2D}: A(q^{-1})y_k(t) = B(q^{-1})r_k(t) + A(q^{-1})y_{k-1}(t) + \Delta w_k(t) \tag{5}$$

where $r_k(t)$ and $y_k(t)$ are, respectively, the input and output of the model, and $\Delta w_k(t) = w_k(t) - w_{k-1}(t)$ is viewed as the disturbance.

Due to the combination of the batch-wise dynamic introduced by ILC law (4), model (5) is a 2D input-output model describing the dynamics of the ILC system. To design updating law $r_k(t)$, two cost functions are introduced, depending on the number of cycles involved.

- Single-batch cost function

$$J(t,k,n_1,n_2) = \sum_{i=1}^{n_1}\alpha(i)\left(\hat{e}_{k|k}(t+i\,|\,t)\right)^2 + \sum_{j=1}^{n_2}\beta(j)\left(r_k(t+j-1)\right)^2 \tag{6}$$

- Multi-batch cost function

$$J(t,k,n_1,n_2,n_3) = \sum_{l=1}^{n_3}\lambda(l)\left(\sum_{i=1}^{n_1}\alpha(i)\left(\hat{e}_{k+l-1|k}(t+i\,|\,t)\right)^2 + \sum_{j=1}^{n_2}\beta(j)\left(r_{k+l-1}(t+j-1)\right)^2\right) \tag{7}$$

where $\hat{e}_{k+l|k}(t+i\,|\,t) = y_r(t+i) - \hat{y}_{k+l|k}(t+i\,|\,t)$ and $\hat{y}_{k+l|k}(t+i\,|\,t)$ represents the estimated output at time $(t+i)$ of the $(k+l)$th cycle based on the measurements before time t of kth cycle, $y_r(t)$, $t = 0,1,...,T$ is the desired trajectory to be tracked, $\alpha(i) \geq 0$, $\beta(j) > 0$ and $\lambda(l) > 0$ are the weighting factors indicating the importance of each cost term, integers $n_1, n_2 (n_1 \geq n_2 > 0)$ are, respectively, referred as the *time-wise prediction horizon* and *control horizon*, and n_3 is called the *batch-wise optimization horizon*.

Obviously, the single-batch cost function (6), where the prediction tracking errors and the updating control effort within specified horizons along one cycle are penalized, is a special case of multi-batch cost function (7) which takes the predicted control performances over several cycles within specified horizons into account. The objective of this paper is to find, at time t of the kth cycle, updating control laws in the MPC framework to minimize the cost functions (6) and (7).

3. 2D-MPILC SCHEMES

3.1 2D prediction model

In MPC framework, the derivation of control scheme requires a prediction model to provide output estimation over the future horizon. According to 2D model (5), at any time t the input and output information of the process can be divided into known and unknown two parts governed by:

$$\begin{aligned}(\boldsymbol{A}_1\ \ \boldsymbol{A}_2)\begin{pmatrix}\boldsymbol{y}_k\left(|_t^{t-n+1}\right)\\ \boldsymbol{y}_k\left(|_{t+n_1}^{t+1}\right)\end{pmatrix} = (\boldsymbol{B}_1\ \ \boldsymbol{B}_2)\begin{pmatrix}\boldsymbol{r}_k\left(|_{t-1}^{t-m+1}\right)\\ \boldsymbol{r}_k\left(|_{t+n_1-1}^{t}\right)\end{pmatrix} + (\boldsymbol{A}_1\ \ \boldsymbol{A}_2)\begin{pmatrix}\boldsymbol{y}_{k-1}\left(|_t^{t-n+1}\right)\\ \boldsymbol{y}_{k-1}\left(|_{t+n_1}^{t+1}\right)\end{pmatrix}\\ +\boldsymbol{\Delta w}_k\left(|_{t+n_1}^{t+1}\right)\end{aligned} \tag{8}$$

where

$$\boldsymbol{f}_k\left(|_{t_2}^{t_1}\right) = \left(f_k(t_1)\ f_k(t_1+1)\ \cdots\ f_k(t_2)\right)^T,\ \boldsymbol{f} \in \{\boldsymbol{y},\boldsymbol{r},\boldsymbol{\Delta w}\} \tag{9}$$

$$(\boldsymbol{A}_1\,|\,\boldsymbol{A}_2) = \left(\begin{array}{ccccc|ccccc} a_n & a_{n-1} & a_{n-2} & \cdots & a_1 & 1 & 0 & \cdots & 0 & 0\\ 0 & a_n & a_{n-1} & \cdots & a_2 & a_1 & 1 & \cdots & 0 & 0\\ 0 & 0 & a_n & \cdots & a_3 & a_2 & a_1 & \cdots & 0 & 0\\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots\\ 0 & 0 & 0 & \cdots & * & * & * & \cdots & a_1 & 1\end{array}\right) \tag{10}$$

$$(\boldsymbol{B}_1\,|\,\boldsymbol{B}_2) = \left(\begin{array}{ccccc|ccccc} b_m & b_{m-1} & b_{m-2} & \cdots & b_2 & b_1 & 0 & \cdots & 0 & 0\\ 0 & b_m & b_{m-1} & \cdots & b_3 & b_2 & b_1 & \cdots & 0 & 0\\ 0 & 0 & b_m & \cdots & b_4 & b_3 & b_2 & \cdots & 0 & 0\\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots\\ 0 & 0 & 0 & \cdots & * & * & * & \cdots & b_2 & b_1\end{array}\right) \tag{11}$$

Since $\boldsymbol{A}_2$ is a nonsingular matrix and $\boldsymbol{\Delta w}_k\left(|_{t+n_1}^{t+1}\right)$ depends on the disturbances of the future, generally assumed to be Gaussian, the best prediction of the outputs over the prediction horizon is therefore

$$\begin{aligned}\hat{\boldsymbol{y}}_{k|k}\left(|_{t+n_1}^{t+1}\,|\,t\right) = \boldsymbol{A}_2^{-1}\boldsymbol{B}_2\boldsymbol{r}_k\left(|_{t+n_1-1}^{t}\right) + \boldsymbol{y}_{k-1}\left(|_{t+n_1}^{t+1}\right) + \boldsymbol{A}_2^{-1}\boldsymbol{B}_1\boldsymbol{r}_k\left(|_{t-1}^{t-m+1}\right)\\ -\boldsymbol{A}_2^{-1}\boldsymbol{A}_1\boldsymbol{y}_k\left(|_t^{t-n+1}\right) + \boldsymbol{A}_2^{-1}\boldsymbol{A}_1\boldsymbol{y}_{k-1}\left(|_t^{t-n+1}\right)\end{aligned} \tag{12}$$

Let

$$\boldsymbol{G} = \boldsymbol{A}_2^{-1}\boldsymbol{B}_2, \quad \boldsymbol{H}_k(t) = \boldsymbol{y}_{k-1}\left(|_{t+n_1}^{t+1}\right) + \boldsymbol{F}_k(t) \tag{13}$$

$$\boldsymbol{F}_k(t) = \boldsymbol{A}_2^{-1}\boldsymbol{B}_1\boldsymbol{r}_k\left(|_{t-1}^{t-m+1}\right) - \boldsymbol{A}_2^{-1}\boldsymbol{A}_1\boldsymbol{y}_k\left(|_t^{t-n+1}\right) + \boldsymbol{A}_2^{-1}\boldsymbol{A}_1\boldsymbol{y}_{k-1}\left(|_t^{t-n+1}\right) \tag{14}$$

Prediction model (12) can be rewritten as

$$\begin{aligned}\hat{\boldsymbol{y}}_{k|k}\left(|_{t+n_1}^{t+1}\,|\,t\right) &= \boldsymbol{G}\boldsymbol{r}_k\left(|_{t+n_1-1}^{t}\right) + \boldsymbol{H}_k(t)\\ &= \boldsymbol{G}\boldsymbol{r}_k\left(|_{t+n_1-1}^{t}\right) + \boldsymbol{y}_{k-1}\left(|_{t+n_1}^{t+1}\right) + \boldsymbol{F}_k(t)\end{aligned} \tag{15}$$

Note that $\boldsymbol{H}_k(t)$ and $\boldsymbol{F}_k(t)$ depend on the available input and output information of current cycle and last cycle. If $\boldsymbol{r}_k\left(\big|_{t+n_1-1}^{t}\right)=\boldsymbol{0}$, one has

$$\boldsymbol{H}_k(t)=\hat{\boldsymbol{y}}_{k|k}\left(\big|_{t+n_1}^{t+1}|\,t\right),\quad \boldsymbol{F}_k(t)=\hat{\boldsymbol{y}}_{k|k}\left(\big|_{t+n_1}^{t+1}|\,t\right)-\boldsymbol{y}_{k-1}\left(\big|_{t+n_1}^{t+1}\right) \quad (16)$$

which show that $\boldsymbol{H}_k(t)$ is the estimation of the system response over the prediction horizon when the control input is not updated, while $\boldsymbol{F}_k(t)$ represents the estimated variation of outputs along the cycle direction.

Prediction model (15) is a 2D model which can be directly extended to estimate the outputs in the following batches, that is,

$$\begin{aligned}\hat{\boldsymbol{y}}_{k+l|k}\left(\big|_{t+n_1}^{t+1}|\,t\right)&=\boldsymbol{G}\boldsymbol{r}_{k+l}\left(\big|_{t+n_1-1}^{t}\right)+\hat{\boldsymbol{H}}_{k+l|k}(t)\\&=\boldsymbol{G}\boldsymbol{r}_{k+l}\left(\big|_{t+n_1-1}^{t}\right)+\hat{\boldsymbol{y}}_{k+l-1|k}\left(\big|_{t+n_1}^{t+1}|\,t\right)+\hat{\boldsymbol{F}}_{k+l|k}(t)\end{aligned} \quad (17)$$

where

$$\hat{\boldsymbol{H}}_{k+l|k}(t)=\hat{\boldsymbol{y}}_{k+l-1|k}\left(\big|_{t+n_1}^{t+1}|\,t\right)+\hat{\boldsymbol{F}}_{k+l|k}(t) \quad (18)$$

$$\begin{aligned}\hat{\boldsymbol{F}}_{k+l|k}(t)=&\boldsymbol{A}_2^{-1}\boldsymbol{B}_1\boldsymbol{r}_k\left(\big|_{t-1}^{t-m+1}\right)-\boldsymbol{A}_2^{-1}\boldsymbol{A}_1\hat{\boldsymbol{y}}_{k+l|k}\left(\big|_{t}^{t-n+1}|\,t\right)\\&+\boldsymbol{A}_2^{-1}\boldsymbol{A}_1\hat{\boldsymbol{y}}_{k+l-1|k}\left(\big|_{t}^{t-n+1}|\,t\right)\end{aligned} \quad (19)$$

If it is assumed that the batch-wise steady control performance have been achieved before time t in the $(k+l)$th ($l>0$) cycle, in other word, that $\boldsymbol{r}_{k+l}\left(\big|_{t-1}^{0}\right)=\boldsymbol{0}$, leading to $\hat{\boldsymbol{F}}_{k+l|k}(t)=\boldsymbol{0}$, then, from the prediction model (17), the following simplified prediction model can be obtained

$$\hat{\boldsymbol{y}}_{k+l|k}\left(\big|_{t+n_1}^{t+1}|\,t\right)=\boldsymbol{G}\boldsymbol{r}_{k+l}\left(\big|_{t+n_1-1}^{t}\right)+\hat{\boldsymbol{y}}_{k+l-1|k}\left(\big|_{t+n_1}^{t+1}|\,t\right),\ l>0 \quad (20)$$

Models (15) and (20) are suitable for the output estimation when the prediction horizon and control horizon are equal, otherwise, the following modifications should be given to matrix $\boldsymbol{G}$ to accommodate the specified configuration of input signal $\boldsymbol{r}_k(\big|_{t+n_1-1}^{t+n_2})$:

- If $r_k(t+i)=0$ for $i=n_2,\dots,n_1-1$, then the last n_1-n_2 columns of matrix $\boldsymbol{G}$ are deleted;
- If $r_k(t+i)=r_k(t+n_2-1)$ for $i=n_2,\dots,n_1-1$, then the last n_1-n_2 columns of matrix $\boldsymbol{G}$ are added to the n_1th column.

Now, the 2D prediction models (15) and (20) are generalized as follows

$$\hat{\boldsymbol{y}}_{k|k}\left(\big|_{t+n_1}^{t+1}|\,t\right)=\boldsymbol{G}\boldsymbol{r}_k\left(\big|_{t+n_2-1}^{t}\right)+\boldsymbol{y}_{k-1}\left(\big|_{t+n_1}^{t+1}\right)+\boldsymbol{F}_k(t) \quad (21)$$

$$\hat{\boldsymbol{y}}_{k+l|k}\left(\big|_{t+n_1}^{t+1}|\,t\right)=\boldsymbol{G}\boldsymbol{r}_{k+l}\left(\big|_{t+n_1-1}^{t}\right)+\hat{\boldsymbol{y}}_{k+l-1|k}\left(\big|_{t+n_1}^{t+1}|\,t\right),\ l>0 \quad (22)$$

In the next subsection, the 2D-MPILC schemes will be derived based on these prediction models.

3.2 2D-MPILC schemes

- Single-batch 2D-MPILC scheme

The cost function (6) can be written in a matrix form

$$\begin{aligned}&J(t,k,n_1,n_2)\\&=\hat{\boldsymbol{e}}_{k|k}^T\left(\big|_{t+n_1}^{t+1}|\,t\right)\boldsymbol{Q}\hat{\boldsymbol{e}}_{k|k}\left(\big|_{t+n_1}^{t+1}|\,t\right)+\boldsymbol{r}_k^T\left(\big|_{t+n_2-1}^{t}\right)\boldsymbol{R}\boldsymbol{r}_k\left(\big|_{t+n_2-1}^{t}\right)\end{aligned} \quad (23)$$

where $\hat{\boldsymbol{e}}_{k|k}\left(\big|_{t+n_1}^{t+1}|\,t\right)=\boldsymbol{y}_r\left(\big|_{t+n_1}^{t+1}\right)-\hat{\boldsymbol{y}}_{k|k}\left(\big|_{t+n_1}^{t+1}|\,t\right)$, and

$$\boldsymbol{Q}=diag\{\alpha(1),\alpha(2),\dots,\alpha(n_1)\} \quad (24)$$

$$\boldsymbol{R}=diag\{\beta(1),\beta(2),\dots,\beta(n_2)\} \quad (25)$$

Clearly, $\boldsymbol{R}>0$, $\boldsymbol{Q}\geq 0$.

It follows from prediction model (21) and optimization algorithm that the cost function (23) is minimized by the following optimal control

$$\boldsymbol{r}_k^*\left(\big|_{t+n_2-1}^{t}\right)=(\boldsymbol{R}+\boldsymbol{G}^T\boldsymbol{Q}\boldsymbol{G})^{-1}\boldsymbol{G}^T\boldsymbol{Q}\left(\boldsymbol{y}_r\left(\big|_{t+n_1}^{t+1}\right)-\boldsymbol{H}_k(t)\right) \quad (26)$$

The positivity of matrix $\boldsymbol{R}$ guarantees the nonsingularity of matrix $\boldsymbol{R}+\boldsymbol{G}^T\boldsymbol{Q}\boldsymbol{G}$. Now, let $\boldsymbol{K}$ be the first row of matrix $(\boldsymbol{R}+\boldsymbol{G}^T\boldsymbol{Q}\boldsymbol{G})^{-1}\boldsymbol{G}^T\boldsymbol{Q}$, the single-batch 2D-MPILC scheme is obtained from definition (13), that is

$$\begin{aligned}\Sigma_{SB-MPILC}:\ u_k(t)&=u_{k-1}(t)+\boldsymbol{K}\left(\boldsymbol{y}_r\left(\big|_{t+n_1}^{t+1}\right)-\boldsymbol{H}_k(t)\right)\\&=u_{k-1}(t)+\boldsymbol{K}\left(\boldsymbol{e}_{k-1}\left(\big|_{t+n_1}^{t+1}\right)-\boldsymbol{F}_k(t)\right)\end{aligned} \quad (27)$$

where $\boldsymbol{e}_{k-1}\left(\big|_{t+n_1}^{t+1}\right)=\boldsymbol{y}_r\left(\big|_{t+n_1}^{t+1}\right)-\boldsymbol{y}_{k-1}\left(\big|_{t+n_1}^{t+1}\right)$ is the tracking error vector over the prediction horizon. It is noted from the first equality that the control signal will not be refined if the model-based estimation of the system response over the prediction horizon is the same as desired trajectory.

- Multi-batch 2D-MPILC scheme

Rewrite multi-batch cost function (7) in the following matrix form

$$\begin{aligned}J(t,k,n_1,n_2,n_3)=\sum_{l=1}^{n_3}\lambda(l)\Big(&\hat{\boldsymbol{e}}_{k+l-1|k}^T\left(\big|_{t+n_1}^{t+1}|\,t\right)\boldsymbol{Q}\hat{\boldsymbol{e}}_{k+l-1|k}\left(\big|_{t+n_1}^{t+1}|\,t\right)\\&+\boldsymbol{r}_{k+l-1}^T\left(\big|_{t+n_2-1}^{t}\right)\boldsymbol{R}\boldsymbol{r}_{k+l-1}\left(\big|_{t+n_2-1}^{t}\right)\Big)\end{aligned} \quad (28)$$

where $\hat{\boldsymbol{e}}_{k+l-1|k}\left(\big|_{t+n_1}^{t+1}|\,t\right)=\boldsymbol{y}_r\left(\big|_{t+n_1}^{t+1}\right)-\hat{\boldsymbol{y}}_{k+l-1|k}\left(\big|_{t+n_1}^{t+1}|\,t\right)$ representing the prediction of the tracking error over the prediction horizon, matrices $\boldsymbol{Q}$ and $\boldsymbol{R}$ are defined by (24) and (25), respectively.

To derive the multi-batch 2D-MPILC scheme, the batch-wise *dynamic programming* will be conducted. Firstly, consider the cost function of the last cycle in the batch-wise optimization horizon

$$\begin{aligned}J(t,k+n_3-1,n_1,n_2)=&\lambda(n_3)\hat{\boldsymbol{e}}_{k+n_3-1|k}^T\left(\big|_{t+n_1}^{t+1}|\,t\right)\boldsymbol{Q}\hat{\boldsymbol{e}}_{k+n_3-1|k}\left(\big|_{t+n_1}^{t+1}|\,t\right)\\&+\lambda(n_3)\boldsymbol{r}_{k+n_3-1}^T\left(\big|_{t+n_2-1}^{t}\right)\boldsymbol{R}\boldsymbol{r}_{k+n_3-1}\left(\big|_{t+n_2-1}^{t}\right)\end{aligned} \quad (29)$$

It follows from prediction model (22) that the above cost function is minimized by the following optimal control

$$\begin{aligned}&\boldsymbol{r}_{k+n_3-1}^*\left(\big|_{t+n_2-1}^{t}\right)\\&=\left(\lambda(n_3)\boldsymbol{R}+\boldsymbol{G}^T\lambda(n_3)\boldsymbol{Q}\boldsymbol{G}\right)^{-1}\boldsymbol{G}^T\lambda(n_3)\boldsymbol{Q}\hat{\boldsymbol{e}}_{k+n_3-2|k}\left(\big|_{t+n_1}^{t+1}|\,t\right)\end{aligned} \quad (30)$$

with the minimal cost defined by

$$\begin{aligned}J^*(t,k+n_3-1,n_1,n_2)=&\lambda(n_3)\left(\hat{\boldsymbol{e}}_{k+n_3-2|k}\left(\big|_{t+n_1}^{t+1}|\,t\right)\right)^T\\&\bullet\left(\boldsymbol{Q}+\boldsymbol{Q}\boldsymbol{G}\left(\boldsymbol{R}+\boldsymbol{G}^T\boldsymbol{Q}\boldsymbol{G}\right)^{-1}\boldsymbol{G}^T\boldsymbol{Q}\right)\bullet\left(\hat{\boldsymbol{e}}_{k+n_3-2|k}\left(\big|_{t+n_1}^{t+1}|\,t\right)\right)\end{aligned} \quad (31)$$

Let $\boldsymbol{P}_{n_3-1}=\lambda(n_3)\left(\boldsymbol{Q}+\boldsymbol{Q}\boldsymbol{G}\left(\boldsymbol{R}+\boldsymbol{G}^T\boldsymbol{Q}\boldsymbol{G}\right)^{-1}\boldsymbol{G}^T\boldsymbol{Q}\right)$, and from the principle of optimality, the minimal cost over the last two cycles within the batch-wise optimization horizon is

$$\begin{aligned}&J^*(t,k+n_3-2,n_1,n_2,2)\\&=\min_{\substack{\boldsymbol{r}_{k+n_3-2}\left(\big|_{t+n_2-1}^{t}\right)\\ \boldsymbol{r}_{k+n_3-1}\left(\big|_{t+n_2-1}^{t}\right)}}\sum_{l=n_3-1}^{n_3}\lambda(l)\Big(\hat{\boldsymbol{e}}_{k+l-1|k}^T\left(\big|_{t+n_1}^{t+1}|\,t\right)\boldsymbol{Q}\hat{\boldsymbol{e}}_{k+l-1|k}\left(\big|_{t+n_1}^{t+1}|\,t\right)\\&\qquad+\boldsymbol{r}_{k+l-1}^T\left(\big|_{t+n_2-1}^{t}\right)\boldsymbol{R}\boldsymbol{r}_{k+l-1}\left(\big|_{t+n_2-1}^{t}\right)\Big)\\&=\min_{\boldsymbol{r}_{k+n_3-2}\left(\big|_{t+n_2-1}^{t}\right)}\lambda(n_3-1)\Big(\hat{\boldsymbol{e}}_{k+n_3-2|k}^T\left(\big|_{t+n_1}^{t+1}|\,t\right)\boldsymbol{Q}\hat{\boldsymbol{e}}_{k+n_3-2|k}\left(\big|_{t+n_1}^{t+1}|\,t\right)\\&\qquad+\boldsymbol{r}_{k+n_3-2}^T\left(\big|_{t+n_2-1}^{t}\right)\boldsymbol{R}\boldsymbol{r}_{k+n_3-2}\left(\big|_{t+n_2-1}^{t}\right)\Big)+J^*(t,k+n_3-1,n_1,n_2)\end{aligned}$$

$$= \min_{\boldsymbol{r}_{k+n_3-2}\left(|_{t+n_2-1}^{t}\right)} \left(\hat{\boldsymbol{e}}^T_{k+n_3-2|k}\left(|_{t+n_1}^{t+1}|\, t\right)\left(\lambda(n_3-1)\boldsymbol{Q}+\boldsymbol{P}_{n_3-1}\right)\hat{\boldsymbol{e}}_{k+n_3-2|k}\left(|_{t+n_1}^{t+1}|\, t\right) + \lambda(n_3-1)\boldsymbol{r}^T_{k+n_3-2}\left(|_{t+n_2-1}^{t}\right)\boldsymbol{R}\boldsymbol{r}_{k+n_3-2}\left(|_{t+n_2-1}^{t}\right)\right) \quad (32)$$

Let $Q_{n_3-1} = \lambda(n_3-1)\boldsymbol{Q} + \boldsymbol{P}_{n_3-1}$. Then the optimal control law is given by

$$\boldsymbol{r}^*_{k+n_3-2}\left(|_{t+n_2-1}^{t}\right) = \left(\lambda(n_3-1)\boldsymbol{R}+\boldsymbol{G}^T\boldsymbol{Q}_{n_3-1}\boldsymbol{G}\right)^{-1}\boldsymbol{G}^T\boldsymbol{Q}_{n_3-1}\hat{\boldsymbol{e}}_{k+n_3-3|k}\left(|_{t+n_1}^{t+1}|\, t\right) \quad (33)$$

The above procedure can be repeated backward along the batch index until the $(k+1)$ th cycle, leading to the optimal control law

$$\boldsymbol{r}^*_{k+1}\left(|_{t+n_2-1}^{t}\right) = \left(\lambda(2)\boldsymbol{R}+\boldsymbol{G}^T\boldsymbol{Q}_2\boldsymbol{G}\right)^{-1}\boldsymbol{G}^T\boldsymbol{Q}_2\hat{\boldsymbol{e}}_{k|k}\left(|_{t+n_1}^{t+1}|\, t\right) \quad (34)$$

and minimal cost of last n_3-1 cycles in the batch-wise optimization horizon computed by

$$J^*(t,k+1,n_1,n_2,n_3-1) = \hat{\boldsymbol{e}}^T_{k|k}\left(|_{t+n_1}^{t+1}|\, t\right)\boldsymbol{P}_1\hat{\boldsymbol{e}}_{k|k}\left(|_{t+n_1}^{t+1}|\, t\right) \quad (35)$$

where matrices $\boldsymbol{Q}_2$ and $\boldsymbol{P}_1$ are determined by the following *backward recursive algorithm*

$$\boldsymbol{P}_{n_3} = \boldsymbol{0}, \qquad \boldsymbol{Q}_l = \lambda(l)\boldsymbol{Q} + \boldsymbol{P}_l \quad (36)$$

$$\boldsymbol{P}_l = \boldsymbol{Q}_{l+1} + \boldsymbol{Q}_{l+1}\boldsymbol{G}\left(\lambda(l+1)\boldsymbol{R}+\boldsymbol{G}^T\boldsymbol{Q}_{l+1}\boldsymbol{G}\right)^{-1}\boldsymbol{G}^T\boldsymbol{Q}_{l+1} \quad (37)$$

$$l = n_3, n_3-1, ..., 1$$

Now, for the k th cycle, the following optimal control law is yielded from prediction model (21)

$$\boldsymbol{r}^*_{k}\left(|_{t+n_2-1}^{t}\right) = \left(\lambda(1)\boldsymbol{R}+\boldsymbol{G}^T\boldsymbol{Q}_1\boldsymbol{G}\right)^{-1}\boldsymbol{G}^T\boldsymbol{Q}_1\left(\boldsymbol{e}_{k-1}\left(|_{t+n_1}^{t+1}\right)-\boldsymbol{F}_k(t)\right) \quad (38)$$

where $\boldsymbol{Q}_1$ is also determined by the backward recursive algorithm (36)(37). Let $\boldsymbol{K}$ be the first row of matrix $\left(\lambda(1)\boldsymbol{R}+\boldsymbol{G}^T\boldsymbol{Q}_1\boldsymbol{G}\right)^{-1}\boldsymbol{G}^T\boldsymbol{Q}_1$, then the multi-batch 2D-MPILC scheme is defined

$$\Sigma_{MB-MPILC}: u_k(t) = u_{k-1}(t) + \boldsymbol{K}\left(\boldsymbol{e}_{k-1}\left(|_{t+n_1}^{t+1}\right)-\boldsymbol{F}_k(t)\right) \quad (39)$$

Remark 3.1. Note from (27) and (39) that both single-batch and multi-batch 2D-MPILC laws have same formulation, and, at any time t, the updating laws depend on the input and output information of current cycle before time t, the output information of last cycle before time t and the tracking errors of last cycle over the prediction horizon.

Remark 3.2. Similar to MPC scheme, the computational burdens for both single-batch and multi-batch 2D-MPILC schemes are dependent on the values of n_1, n_2 and n_3, which can be balanced by practitioner in terms of computational load and control performance.

Remark 3.3. Single-batch 2D-MPILC scheme which takes only the control performance of current cycle into account is more suitable for the case when large control errors exist in the last cycle, such as the first cycle of the batch process. Multi-batch 2D-MPILC scheme which takes the control performances of several cycles into account can provides faster convergence along the batch index.

Remark 3.4. The proposed design methods can be extended to general cases of MIMO time-varying linear processes with input time delay. Proper selection of time-varying weighting matrices $\boldsymbol{Q}$ and $\boldsymbol{R}$ can also be used to accommodate time-varying dynamic characteristics of the processes.

4. STRUCTURE ANALYSIS

In a 2D system view, the resulted closed-loop system is composed of a 2D process Σ_{2D} and a controller $\Sigma_{SB-MPILC}$ or $\Sigma_{MB-MPILC}$, as shown in Figure 1(a), where the dot-arrow lines indicate the information flows of the last cycle from the storages, while the solid-arrow lines indicate the real-time information flows. Plant Σ_{2D} is a 2D system consisting of a batch process Σ_{BP} and an iteration loop, while $\Sigma_{SB-MPILC}$ or $\Sigma_{MB-MPILC}$ is a 2D model predictive control scheme.

In an ILC system view, ILC law (27) or (39) can be reformulated as

$$u_k(t) = u_{k-1}(t) + \boldsymbol{K}\left(\boldsymbol{I} \quad \boldsymbol{A}_2^{-1}\boldsymbol{A}_1 \quad -\boldsymbol{A}_2^{-1}\boldsymbol{B}_1\right)\begin{pmatrix}\boldsymbol{e}_{k-1}\left(|_{t+n_1}^{t+1}\right)\\ \boldsymbol{y}_k\left(|_{t}^{t-n+1}\right)-\boldsymbol{y}_{k-1}\left(|_{t}^{t-n+1}\right)\\ \boldsymbol{u}_k\left(|_{t-1}^{t-m+1}\right)-\boldsymbol{u}_{k-1}\left(|_{t-1}^{t-m+1}\right)\end{pmatrix} \quad (40)$$

Let $\left(\boldsymbol{K}_1 \quad \boldsymbol{K}_2 \quad \boldsymbol{K}_3\right) = \boldsymbol{K}\left(\boldsymbol{I} \quad \boldsymbol{A}_2^{-1}\boldsymbol{A}_1 \quad -\boldsymbol{A}_2^{-1}\boldsymbol{B}_1\right)$, then the above ILC law can be decomposed as

$$u_k(t) = u_{ilc,k}(t) + u_{mpc,k}(t) \quad (41)$$

where $u_{ilc,k}(t)$ and $u_{mpc,k}(t)$ are described by

$$\Sigma_{ILC}: \quad u_{ilc,k}(t) = u_{ilc,k-1}(t) + \boldsymbol{K}_1\boldsymbol{e}_{k-1}\left(|_{t+n_1}^{t+1}\right) \quad (42)$$

$$\Sigma_{MPC}: \quad u_{mpc,k}(t) = \boldsymbol{K}_2\boldsymbol{y}_k\left(|_{t}^{t-n+1}\right) + \boldsymbol{K}_3\boldsymbol{u}_k\left(|_{t-1}^{t-m+1}\right) \quad (43)$$

It is clear that Σ_{ILC} is an ILC law for the improvement of control performance from batch to batch, while Σ_{MPC} is an MPC law ensuring control performance over time of each cycle. The 2D based design framework gives a united design of these two types of control laws. The equivalent structure of the closed-loop system is shown as Figure 1(b), where the triangular blocks represent the proportional controllers.

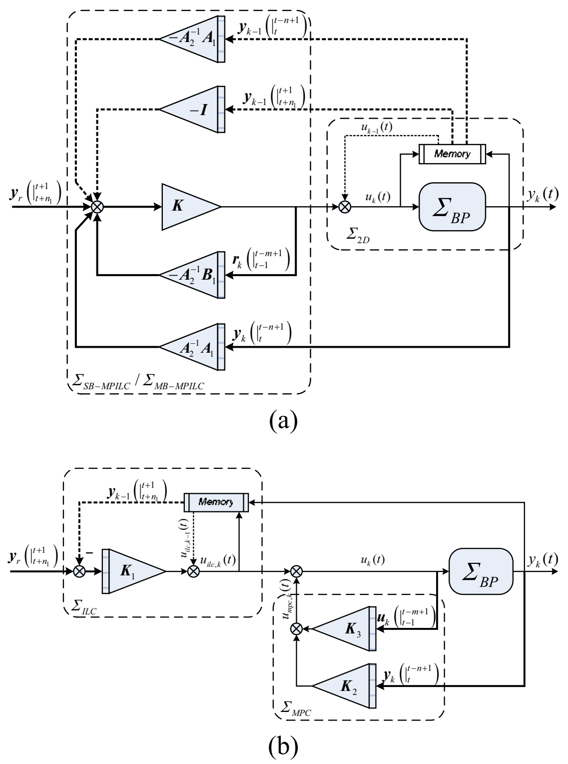

Fig 1. Equivalent system structures: (a) 2D system structure; (b) ILC system.

5. 2D-MPILC SCHEMES BASED ON 2D CONTROL PENALTY

In cost functions (6) and (7), the tracking errors and the change of control variable along the batch index are penalized. Theoretically, for any desired trajectory, the optimal control driving the cost function (6) and (7) approaching to zero can be achieved gradually by refining the control signal iteratively. An ILC strategy, in essence, searches for process input by inversing the process dynamic to generate the desired trajectory. For a non-minimum phase system, however, the inversion of the process is unstable; this can result in that the optimal control may not be physically realizable. One way to solve this problem is to place hard restrictions on the control variable, leading to a quadratic programming problem for the controller design (Lee *et al.*, 2000). Another method is to introduce a penalty to the change of the control variable along the time axis into the cost functions as well, resulting in the following cost functions with 2D control penalty,

- Single-batch cost function

$$J(t,k,n_1,n_2)=\sum_{i=1}^{n_1}\alpha(i)\left(\hat{e}_{k|k}(t+i\,|\,t)\right)^2+\sum_{j=1}^{n_2}\beta(j)\left(r_k(t+j-1)\right)^2 +\sum_{j=1}^{n_2}\gamma(j)\left(\Delta u_k(t+j-1)\right)^2 \qquad (44)$$

- Multi-batch cost function

$$J(t,k,n_1,n_2,n_3) =\sum_{l=1}^{n_3}\lambda(l)\left(\sum_{i=1}^{n_1}\alpha(i)\left(\hat{e}_{k+l-1|k}(t+i\,|\,t)\right)^2+\sum_{j=1}^{n_2}\beta(j)\left(r_{k+l-1}(t+j-1)\right)^2 +\sum_{j=1}^{n_2}\gamma(j)\left(\Delta u_{k+l-1}(t+j-1)\right)^2\right) \qquad (45)$$

where $\Delta u_k(t)=u_k(t)-u_k(t-1)$, indicating the change of the control variable along the time index, $\gamma(j)\geq 0$ is the weighting factor.

In this section, single-batch 2D-MPILC scheme will be developed based on the cost function (44). By using dynamic programming, the multi-batch case can be developed in a similar way.

Clearly, cost function (44) can be written in matrix form as

$$J(t,k,n_1,n_2)=\hat{\boldsymbol{e}}_{k|k}^T\left(\big|_{t+n_1}^{t+1}\,|\,t\right)\boldsymbol{Q}\hat{\boldsymbol{e}}_{k|k}\left(\big|_{t+n_1}^{t+1}\,|\,t\right)+\boldsymbol{r}_k^T\left(\big|_{t+n_2-1}^{t}\right)\boldsymbol{R}\boldsymbol{r}_k\left(\big|_{t+n_2-1}^{t}\right) +\boldsymbol{\Delta u}_k^T\left(\big|_{t+n_2-1}^{t}\right)\boldsymbol{S}\boldsymbol{\Delta u}_k\left(\big|_{t+n_2-1}^{t}\right)$$

$$=\begin{pmatrix}\hat{\boldsymbol{e}}_{k|k}\left(\big|_{t+n_1}^{t+1}\,|\,t\right)\\ \boldsymbol{\Delta u}_k\left(\big|_{t+n_2-1}^{t}\right)\end{pmatrix}^T\begin{pmatrix}\boldsymbol{Q} & \boldsymbol{0}\\ \boldsymbol{0} & \boldsymbol{S}\end{pmatrix}\begin{pmatrix}\hat{\boldsymbol{e}}_{k|k}\left(\big|_{t+n_1}^{t+1}\,|\,t\right)\\ \boldsymbol{\Delta u}_k\left(\big|_{t+n_2-1}^{t}\right)\end{pmatrix}+\boldsymbol{r}_k^T\left(\big|_{t+n_2-1}^{t}\right)\boldsymbol{R}\boldsymbol{r}_k\left(\big|_{t+n_2-1}^{t}\right)$$

where $\boldsymbol{S}=diag\{\gamma(1),\gamma(2),...,\gamma(n_2),\}$. From relation

$$\Delta u_k(t)=r_k(t)-r_k(t-1)+\Delta u_{k-1}(t) \qquad (46)$$

we have

$$\boldsymbol{\Delta u}_k\left(\big|_{t+n_2-1}^{t}\right)=\boldsymbol{H}\boldsymbol{r}_k\left(\big|_{t+n_2-1}^{t}\right)+\boldsymbol{\Delta u}_{k-1}\left(\big|_{t+n_2-1}^{t}\right) \qquad (47)$$

where

$$\boldsymbol{H}=\begin{pmatrix}1 & 0 & 0 & \cdots\\ -1 & 1 & 0 & \cdots\\ 0 & -1 & 1 & \cdots\\ \vdots & \vdots & \vdots & \ddots\end{pmatrix}_{n_2\times n_2} \qquad (48)$$

Together with 2D prediction model (21), the following augmented 2D prediction model is obtained

$$\begin{pmatrix}\hat{\boldsymbol{e}}_{k|k}\left(\big|_{t+n_1}^{t+1}\,|\,t\right)\\ \boldsymbol{\Delta u}_k\left(\big|_{t+n_2-1}^{t}\right)\end{pmatrix}=\begin{pmatrix}-\boldsymbol{G}\\ \boldsymbol{H}\end{pmatrix}\boldsymbol{r}_k\left(\big|_{t+n_2-1}^{t}\right)+\begin{pmatrix}\boldsymbol{e}_{k-1}\left(\big|_{t+n_1}^{t+1}\right)\\ \boldsymbol{\Delta u}_{k-1}\left(\big|_{t+n_2-1}^{t}\right)\end{pmatrix}+\begin{pmatrix}-\boldsymbol{F}_k(t)\\ \boldsymbol{0}\end{pmatrix}$$

Based on optimization algorithm, the optimal control law is obtained as

$$\boldsymbol{r}_k^*\left(\big|_{t+n_2-1}^{t}\right)=-\left(\boldsymbol{R}+\begin{pmatrix}-\boldsymbol{G}\\ \boldsymbol{H}\end{pmatrix}^T\begin{pmatrix}\boldsymbol{Q} & \boldsymbol{0}\\ \boldsymbol{0} & \boldsymbol{S}\end{pmatrix}\begin{pmatrix}-\boldsymbol{G}\\ \boldsymbol{H}\end{pmatrix}\right)^{-1}\begin{pmatrix}-\boldsymbol{G}\\ \boldsymbol{H}\end{pmatrix}^T\begin{pmatrix}\boldsymbol{Q} & \boldsymbol{0}\\ \boldsymbol{0} & \boldsymbol{S}\end{pmatrix} \bullet\left(\begin{pmatrix}\boldsymbol{e}_{k-1}\left(\big|_{t+n_1}^{t+1}\right)\\ \boldsymbol{\Delta u}_{k-1}\left(\big|_{t+n_2-1}^{t}\right)\end{pmatrix}+\begin{pmatrix}-\boldsymbol{F}_k(t)\\ \boldsymbol{0}\end{pmatrix}\right)$$

$$=\left(\boldsymbol{R}+\boldsymbol{G}^T\boldsymbol{Q}\boldsymbol{G}+\boldsymbol{H}^T\boldsymbol{S}\boldsymbol{H}\right)^{-1}\left(\boldsymbol{G}^T\boldsymbol{Q}\left(\boldsymbol{e}_{k-1}\left(\big|_{t+n_1}^{t+1}\right)-\boldsymbol{F}_k(t)\right) -\boldsymbol{H}^T\boldsymbol{S}\boldsymbol{\Delta u}_{k-1}\left(\big|_{t+n_2-1}^{t}\right)\right) \qquad (49)$$

Let $\boldsymbol{K}_1$ and $\boldsymbol{K}_2$ indicate respectively the first row of matrices $\left(\boldsymbol{R}+\boldsymbol{G}^T\boldsymbol{Q}\boldsymbol{G}+\boldsymbol{H}^T\boldsymbol{S}\boldsymbol{H}\right)^{-1}\boldsymbol{G}^T\boldsymbol{Q}$ and $-\left(\boldsymbol{R}+\boldsymbol{G}^T\boldsymbol{Q}\boldsymbol{G}+\boldsymbol{H}^T\boldsymbol{S}\boldsymbol{H}\right)^{-1}\boldsymbol{H}^T\boldsymbol{S}$, then the single-batch 2D-MPILC law can be formulated as

$$u_k(t)=u_{k-1}(t)+\boldsymbol{K}_1\left(\boldsymbol{e}_{k-1}\left(\big|_{t+n_1}^{t+1}\right)-\boldsymbol{F}_k(t)\right)+\boldsymbol{K}_2\boldsymbol{\Delta u}_{k-1}\left(\big|_{t+n_2-1}^{t}\right) \qquad (50)$$

Different from 2D-MPILC law (27), the above control law has an additional term depending on the changes of control signal in last cycle over the prediction horizon. As the change of the control variable along both time and cycle directions are penalized, weighting factors $\beta(\bullet)$ and $\gamma(\bullet)$ should be designed properly to ensure the feasible and necessary variation of control signal along both time and cycle indices.

6. EXAMPLES

Injection molding process is a typical repetitive process (Gao *et al.*, 2001; Shi *et al.*, 2005), where many process variables need to be controlled to follow certain profiles repetitively to ensure the product quality. To illustrate the effectiveness of the proposed schemes, a simulation is performed on the following the injection velocity control process

$$\Sigma_{BP}:P_k(q^{-1})=\frac{2.651q^{-1}+5.298q^{-2}+0.5805q^{-3}}{1-1.454q^{-1}+0.5285q^{-2}-0.04736q^{-3}} \qquad (51)$$

The designs of control schemes are based on the following simplified model

$$\Sigma_{MBP}: \qquad M_k(q^{-1})=q^{-1}\frac{13.81q^{-1}}{1-0.9524q^{-1}} \qquad (52)$$

For cost functions (6) and (7) with parameters

$$n_1=10,\ n_2=10,\ n_3=3,\ \alpha=1, \beta=100, \lambda=1 \qquad (53)$$

single-batch and multi-batch 2D-MPILC laws are designed, and their set-point tracking results are shown in Figure 2 and Figure 3, respectively. In both cases, MPC control law guarantees a good tracking performance even in the first cycle and the control performance improves by the ILC from cycle to cycle. As process (51) has an unstable zero, significant oscillations of control input signal are required for the perfect tracking, as seen in the control signals of the 30th cycle shown in Figure 2 and Figure 3. This may be not practical. To solve this problem, single-batch 2D-MPILC law (50) is designed based on the cost function (44) with weighting factor $\gamma=10$. The simulation results are shown in Figure 4. It is clearly seen that the oscillation of control signal are reduced with the satisfactory control performance maintained. The sum of tacking errors over each cycle for different control schemes are shown in Figure 5, indicating that multi-batch 2D-MPILC scheme has the best convergence along the cycle index.

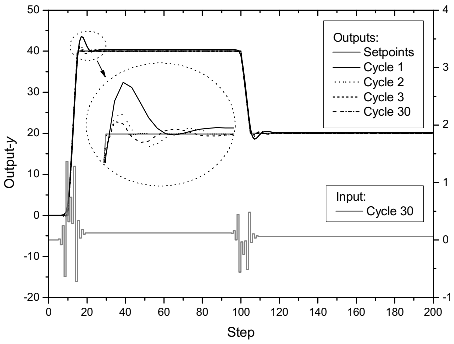

Fig 2. Output responses and control signal of single-batch 2D-MPILC scheme.

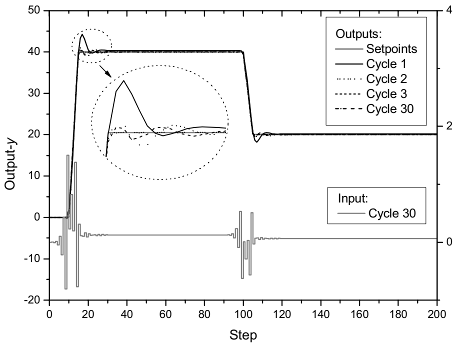

Fig 3. Output responses and control signal of multi-batch 2D-MPILC scheme.

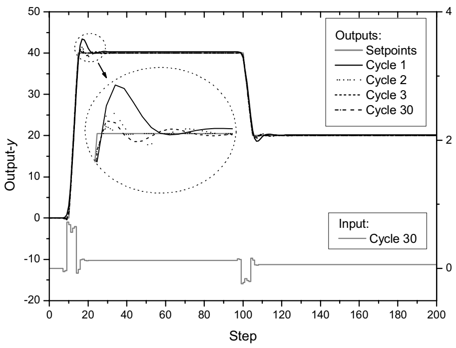

Fig 4. Output responses and control signal of single-batch 2D-MPILC scheme based on 2D control penalty.

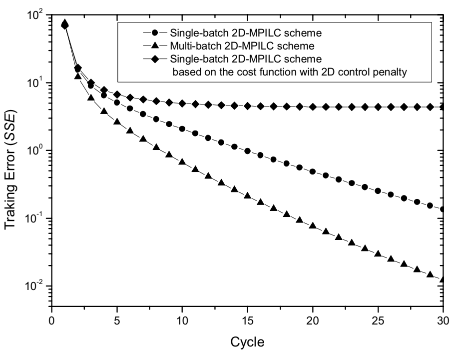

Fig 5. Tracking errors of different control schemes.

7. CONCLUSIONS

In this paper, iterative learning control design problem has been modelled and solved from the 2D system viewpoint. Single-batch and multi-batch 2D-MPILC schemes have been developed in the framework of MPC of a 2D system problem. The resulted 2D-MPILC laws implicitly combine an MPC along time with an ILC along cycle to ensure the optimal control in 2D sense. The computational load and performance of the proposed design methods can be balanced by proper selection of design parameters in the defined cost functions. 2D control penalty can be further introduced to the 2D-MPILC design to ensure non-oscillatory operation.

REFERENCES

Amann, N., Owens, D.H., Rogers, E. (1995). Iterative learning control for discrete time systems using optimal feedback and feedforward actions. *Proceedings of the 34th Conference on Decision and Control, New Orleans, LA*, pp1696-1701.

Amann, N., Owens, D.H., Rogers, E. (1996). Iterative learning control using optimal feedback and feedforward actions. *Int. J. Control*, Vol. 65(2), 277-293.

Arimoto, S., Kawamura, S., Miyazaki, F. (1984). Bettering operation of robots by learning. *Journal of Robotic Systems*, Vol.1(2), pp123-140.

Gao, F., Yang, Y., Shao, C. (2001). Robust iterative learning control with applications to injection molding process. *Chemical Engineering Science*, Vol.56, pp7025-7034.

Geng, Z., Carroll, R., Xie, J. (1990) Two-dimensional model and algorithm analysis for a class of iterative learning control systems. *Int. J. Control*, Vol. 52(4), pp833-862.

Kurek, J.E., Zaremba, M.B. (1993). Iterative learning control synthesis based on 2-D system theory. *IEEE Transactions on Automatic Control*, Vol.38(1), pp121-125.

Lee, J.H., Lee, K.S., Kim, W.C. (2000). Model-based iterative learning control with a quadratic criterion for time-varying linear systems. *Automatica*, Vol.36, pp641-657.

Owens, D.H., Amann, N., Rogers, E., French, M. (2000). Analysis of linear iterative learning control schemes – a 2D system/repetitive processes approach. *Multidimensional Systems and Signal Processing*, Vol.11, pp125-177.

Shi, J., Gao, F., Wu, T.-J. (2005). Robust design of integrated feedback and iterative learning control of a batch process based on a 2D Roesser system. *Journal of Process Control*, Vol.15, pp907-924.

Xu, J.-X., Bien, Z.Z. (1998). The frontiers of iterative learning control. In: *Iterative Learning Control: Analysis, Design, Integration and Application.* Kluwer Academic Publishers, Boston/Dordrecht/London, pp9-35.