A TWO-STAGE ALGORITHM FOR COMBINED ITERATIVE LEARNING CONTROL WITH REAL-TIME FEEDBACK; A STATE SPACE FORMULATION

Insik Chin * **Moonki Cho** ** **S. Joe Qin** *** **Kwang S. Lee** ****,[1]

* *Samsung Electronics, Keeheung, Kyunggido, Korea*
** *Conwell Ltd. Co., Seoul, Korea*
*** *Dept. Chem. Engng., The Univ. of Texas at Austin, Austin, TX 78712-1062, USA*
**** *Dept. Chem. and Biomol. Engng., Sogang Univ. 1-Shinsoodong, Mapogu, Seoul 121-742, Korea*

Abstract: A new combined iterative learning control (ILC) and real-time feedback control (RFC) algorithm has been proposed on the basis of the state space formulation and the two-stage implementation. The proposed method assumes Gaussian disturbances and deals with the batch-wise recurrent and nonrecurrent disturbances by independent LQG formulation for ILC and RFC, respectively. In this way, the problem with the existing combined ILC-RFC methods that the nonrecurrent real-time disturbance causees ILC to digress from its convergence track along the run index could be overcome. The performance of the proposed technique has been demonstrated using numerical simulation.

Keywords: Iterative Learning Control, Run-to-run Control, Batch Process Control, LQG Control, Disturbance Rejection

1. INTRODUCTION

Iterative learning control (ILC) is a special batch process control technique which is concerned with the issue of learning from the past operations with an aim to attain the ultimate tracking performance under model uncertainty and run-wise recurrent (RWR) disturbances. In practical applications, ILC alone is seldom used but real-time feedback control (RFC) is combined to treat run-wise nonrecurrent (RWNR) real-time disturbances. The combination of RFC with ILC (RFC-ILC) is typically done(Hashimoto and Xu, 1987) (Xu *et al.*, 1995)(Moore, 1999)(Phan *et al.*, 2000) as a feedforward-feedback configuration such as

$$u_k(t) = u_{k-1}(t) + He_{k-1}(1:N) + Fe_k(1:t) \quad (1)$$

where e_k is the control error at the k^{th} run; $(i:j)$ means data from $t = i$ to j; H and F represent the (possibly time-varying) gains for ILC and RFC, respectively. Phan *et al.* (2000) have carried out a comprehensive study on the ILC structure in a deterministic setting and found that the existing RFC-ILC methods can be unified by the above formulation.

In practical applications, various disturbances with both RWR and RWNR nature may enter the

[1] All correspondence should be addressed to: phone +82-2-705-8477, fax +82-2-3272-0319, e-mail kslee@sogang.ac.kr

process and the RFC-ILC methods based on (1) may show performance limitations. For example, when a large RWNR disturbance occurs, the RFC action $H_2(t)e_k(1:t)$ may produce a large input change. This leads $u_k(\cdot)$ to digress from the convergence track for learning and deteriorates the performance of the subsequent runs. The most desirable controller action is that RFC responds only to RWNR real-time disturbances whereas ILC reacts only to RWR disturbances. It is evident that (1) lacks such feature.

Considering the above problems, the authors have recently developed the so-called two-stage batch control (TBC) technique and proposed an impulse response model-based algorithm(Chin, 2004). The TBC technique is based on a framework to update ILC in a separate track from RFC and thus can prevent ILC from being affected by RFC.

The purpose of the present paper is to extend the previous TBC technique to a stochastic state space model-based algorithm. For this, a disturbance model is first assumed by decomposing the disturbance into three parts: the RWR part, the RWNR part, and the measurement noise. Also the input is split into two parts: one for ILC and the other for RFC. With the above decomposition, we devised an LQG-based two-stage control technique where RFC and ILC are executed in turn during and after a batch run. During this procedure, the effect of the real-time RWNR disturbance is appropriately discriminated from that of the RWR disturbance and the influence of the real-time disturbance is effectively prevented from being carried over to future runs.

2. DISTURBANCE PROPAGATION IN THE EXISTING TECHNIQUES

In this section, we first revisit how the existing ILC and RFC-ILC techniques respond to the real-time RWNR disturbance(Chin, 2004).

2.1 Process Modelling

We consider a linear discrete-time batch process with $u_k(t)$, $y_k(t)$, and $d_k(t)$ as input, output, and disturbance at the k^{th} run, respectively, defined over a finite interval with N sampling steps. Such a process can be represented by a linear algebraic system between the input and output sequence vectors over the underlying discrete-time domain.

$$\mathbf{y}_k = \mathbf{G}\mathbf{u}_k - \mathbf{d}_k \tag{2}$$

where

$$\begin{aligned}\mathbf{u}_k &= \left[u_k^T(0)\ u_k^T(1)\cdots u_k^T(N-1)\right]^T \\ \mathbf{y}_k &= \left[y_k^T(1)\ y_k^T(2)\cdots y_k^T(N)\right]^T\end{aligned} \tag{3}$$

and likewise for $\mathbf{d}_k$. In the above, $\mathbf{G}$ is a low-triangular matrix whose respective columns consist of (possibly time-varying) pulse response coefficients. The disturbance can be decomposed as

$$\mathbf{d}_k = \mathbf{s}_k + \mathbf{v}_k \tag{4}$$

where $\mathbf{s}_k$ and $\mathbf{v}_k$ represent the RWR and RWNR parts, respectively.

If we represent the RWR part as an integrated white noise process along the run index, then $\mathbf{d}_k$ can be expressed as follows:

$$\begin{aligned}\mathbf{s}_k &= \mathbf{s}_{k-1} + \mathbf{w}_k \\ \mathbf{d}_k &= \mathbf{w}_k + \mathbf{v}_k\end{aligned} \tag{5}$$

where both $\{\mathbf{w}_k\}$ and $\{\mathbf{v}_k\}$ represent the zero-mean white noise processes along k.

Let $\mathbf{e}_k \triangleq \mathbf{y}_d - \mathbf{y}_k$ and $\bar{\mathbf{e}}_k \triangleq \mathbf{e}_k - \mathbf{v}_k$ where $\mathbf{y}_d$ is the desired output trajectory. Then the following inter-run transition model of tracking error can be derived from (2) and (5):

$$\begin{aligned}\bar{\mathbf{e}}_k &= \bar{\mathbf{e}}_{k-1} - \mathbf{G}\Delta\mathbf{u}_k + \mathbf{w}_k \\ \mathbf{e}_k &= \bar{\mathbf{e}}_k + \mathbf{v}_k.\end{aligned} \tag{6}$$

where $\Delta\mathbf{u}_k \triangleq \mathbf{u}_k - \mathbf{u}_{k-1}$.

2.2 Pure Iterative Learning Control

The pure ILC algorithm can be written as

$$\Delta\mathbf{u}_k = \mathbf{H}\bar{\mathbf{e}}_{k-1} \tag{7}$$

with an appropriate learning gain $\mathbf{H}$. In practice, $\bar{\mathbf{e}}_{k-1}$ should be replaced by an estimate.

Substituting (7) into (6) gives

$$\begin{aligned}\bar{\mathbf{e}}_k &= [\mathbf{I} - \mathbf{G}\mathbf{H}]\bar{\mathbf{e}}_{k-1} + \mathbf{w}_k \\ \mathbf{e}_k &= [\mathbf{I} - \mathbf{G}\mathbf{H}]\bar{\mathbf{e}}_{k-1} + \mathbf{v}_k + \mathbf{w}_k\end{aligned} \tag{8}$$

It can be seen that $\Delta\mathbf{u}_{k+1}$ is not affected by $\mathbf{v}_k$, which implies that ILC based on (7) can keep its integrity from the effect of the RWNR real-time disturbance. If $\mathbf{e}_{k-1}$ is used in (7) instead of $\bar{\mathbf{e}}_{k-1}$, $\Delta\mathbf{u}_{k+1}$ is affected by $\mathbf{v}_k$.

2.3 Iterative Learning Control with Real-time Feedback

The feedforward-feedback combination of RFC and ILC can be expressed in a general form as

$$\Delta \mathbf{u}_k = \mathbf{H}\bar{\mathbf{e}}_{k-1} + \mathbf{F}\mathbf{e}_k \qquad (9)$$

To reject the real-time disturbance, $\mathbf{e}_k$ instead of $\bar{\mathbf{e}}_k$ is considered in the current run.

Substituting (9) into (6) results in

$$\begin{aligned}\mathbf{e}_k &= [\mathbf{I}+\mathbf{G}\mathbf{F}]^{-1}([\mathbf{I}-\mathbf{G}\mathbf{H}]\bar{\mathbf{e}}_{k-1}+\mathbf{v}_k+\mathbf{w}_k) \qquad (10)\\ \bar{\mathbf{e}}_k &= [\mathbf{I}+\mathbf{G}\mathbf{F}]^{-1}([\mathbf{I}-\mathbf{G}\mathbf{H}]\bar{\mathbf{e}}_{k-1}-\mathbf{G}\mathbf{F}\mathbf{v}_k+\mathbf{w}_k)\end{aligned}$$

It can seen that $\mathbf{v}_k$ has an effect on $\bar{\mathbf{e}}_k$ and, consequently, $\Delta\mathbf{u}_{k+1}$ is affected by the RWNR disturbance.

3. A NEW RFC-ILC FRAMEWORK

From the aforementioned investigation, the present RFC-ILC configuration cannot maintain the integrity of the learning procedure. Occurrence of the RWNR disturbance interferes with the input signal updating. To fix the problem, Chin *et al.*(2004) proposed to separate the input for ILC from $\mathbf{u}_k$ and to let ILC proceed independently of the RWNR disturbance. For this, it is necessary to recast the model in (6) first.

3.1 Process Modeling

For more complete handling of the disturbance, we decompose the disturance into three terms: $\mathbf{s}_k$, $\mathbf{v}_k$, and $\mathbf{n}_k$ which refer to the RWR and RWNR disturbances, and measurement noise, respectively.

$$\begin{aligned}\mathbf{d}_k &= \mathbf{s}_k + \mathbf{v}_k + \mathbf{n}_k \qquad (11)\\ \mathbf{s}_k &= \mathbf{s}_{k-1} + \mathbf{w}_k.\end{aligned}$$

Also we decompose $\mathbf{u}_k$ into $\bar{\mathbf{u}}_k$ and $\hat{\mathbf{u}}_k$ such that $\mathbf{u} = \bar{\mathbf{u}} + \hat{\mathbf{u}}$, which correspond to the ILC and RFC parts, respectively. Then the process model can be expressed as

$$\begin{aligned}\mathbf{y}_k &= \mathbf{G}\mathbf{u}_k - \mathbf{d}_k = \mathbf{G}(\bar{\mathbf{u}}_k + \hat{\mathbf{u}}_k) - (\mathbf{s}_k + \mathbf{v}_k + \mathbf{n}_k)\\ &= \underbrace{\mathbf{G}\bar{\mathbf{u}}_k - \mathbf{s}_k}_{\bar{\mathbf{y}}_k} + \mathbf{G}\hat{\mathbf{u}}_k - \mathbf{v}_k - \mathbf{n}_k\\ &= \underbrace{\bar{\mathbf{y}}_k + \mathbf{G}\hat{\mathbf{u}}_k - \mathbf{v}_k}_{\hat{\mathbf{y}}_k} - \mathbf{n}_k = \hat{\mathbf{y}}_k - \mathbf{n}_k \qquad (12)\end{aligned}$$

Similarly to (6), the following model equation can be derived from (12):

$$\begin{aligned}\bar{\mathbf{e}}_k &= \bar{\mathbf{e}}_{k-1} - \mathbf{G}\Delta\bar{\mathbf{u}}_k + \mathbf{w}_k\\ \hat{\mathbf{e}}_k &= \bar{\mathbf{e}}_k - \mathbf{G}\hat{\mathbf{u}}_k + \mathbf{v}_k \qquad (13)\\ \mathbf{e}_k &= \hat{\mathbf{e}}_k + \mathbf{n}_k\end{aligned}$$

where $\bar{\mathbf{e}}_k \triangleq \mathbf{y}_d - \bar{\mathbf{y}}_k$ and $\hat{\mathbf{e}}_k \triangleq \mathbf{y}_d - \hat{\mathbf{y}}_k$.

3.2 Input Updating Law

For (13), one may consider the following ILC and RFC laws, respectively:

$$\begin{aligned}\Delta\bar{\mathbf{u}}_k &= \mathbf{H}_1\bar{\mathbf{e}}_{k-1} \;\rightarrow\; \bar{\mathbf{u}}_k = \bar{\mathbf{u}}_{k-1} + \mathbf{H}_1\bar{\mathbf{e}}_{k-1} \qquad (14)\\ \hat{\mathbf{u}}_k &= \mathbf{H}_2\hat{\mathbf{e}}_k \;\rightarrow\; \mathbf{u}_k = \bar{\mathbf{u}}_k + \mathbf{H}_2\hat{\mathbf{e}}_k\end{aligned}$$

Substitution of the above equations into (13) yields

$$\begin{aligned}\bar{\mathbf{e}}_k &= [\mathbf{I} - \mathbf{G}\mathbf{H}_1]\bar{\mathbf{e}}_{k-1} + \mathbf{w}_k \qquad (15)\\ \mathbf{e}_k &= [\mathbf{I} + \mathbf{G}\mathbf{H}_2]^{-1}(\bar{\mathbf{e}}_k + \mathbf{v}_k) + \mathbf{n}_k.\end{aligned}$$

The above is what we have desired. $\mathbf{v}_k$ doesn't affect $\bar{\mathbf{e}}_k$ and is attenuated in $\mathbf{e}_k$ by the separate RFC action. In real implementations, $\bar{\mathbf{e}}_k$ and $\hat{\mathbf{e}}_k$ cannot be directly measured but need to be estimated using appropriate observers.

Implementation of the above method should be done in two stages as can be noticed from (14). After finishing the $k-1^{\text{th}}$ run, $\bar{\mathbf{u}}_k$ is calculated. With $\bar{\mathbf{u}}_k$ as the bias input signal, $\hat{u}_k(t)$ is computed and $u_k(t)$ is implemented to the process in real-time during the k^{th} run.

4. STATE SPACE FORMULATION OF THE TWO-STAGE RFC-ILC ALGORITHM

4.1 Recasting of State Space Model

Suppose that the batch process dynamics is described by the following stochastic state space model:

$$\begin{aligned}x(t+1) &= Ax(t) + Bu(t) + w(t)\\ y(t) &= Cx(t) + v(t), \quad t \in \{0\cdots,N-1\} \qquad (16)\end{aligned}$$

In the above, $w(t)$ and $v(t)$ refer to zero-mean white noise sequences in time but they may show RWR behavior. In fact, both $w_k(t)$ and $v_k(t)$ may exhibit drifting behavior along k in addition to random fluctuations. Such behavior can be reasonably modeled by the equation

$$\begin{aligned}w_k(t) &= \bar{w}_k(t) + \hat{w}_k(t), \;\; \bar{w}_k - \bar{w}_{k-1}(t) = m_k(t)\\ v_k(t) &= \bar{v}_k(t) + \hat{v}_k(t), \;\; \bar{v}_k - \bar{v}_{k-1}(t) = n_k(t) \qquad (17)\end{aligned}$$

where $\hat{w}_k(t)$, $m_k(t)$, $\hat{v}_k(t)$, and $n_k(t)$ are zero-mean white noise sequences in both k and t. In the above, $\bar{w}_k(t)$ and $\bar{v}_k(t)$ represent the parts of the disturbances that ILC rejects; $\hat{w}_k(t)$ is the one that RFC handles; $\hat{v}_k(t)$ refers to the measurement noise that should be filtered, respectively.

Now, we decompose $u_k(t)$ into $\bar{u}_k(t)$ and $\hat{u}_k(t)$, and decompose (16) into two parts, one that is driven by $\bar{u}_k(t)$, $\bar{w}_k(t)$, and $\bar{v}_k(t)$, and the other by

$\hat{u}_k(t)$ and $\hat{v}_k(t)$. Rearrangement of the first part after introducing the models for $\bar{w}_k(t)$ and $\bar{v}_k(t)$, the two parts are represented as

$$\begin{aligned} \bar{x}_k(t+1) &= A\bar{x}_k(t) + B\Delta\bar{u}_k(t) + m_k(t) \\ \bar{y}_k(t) &= C\bar{x}_k(t) + \bar{y}_{k-1}(t) + n_k(t) \end{aligned} \quad (18)$$

$$\begin{aligned} \hat{x}_k(t+1) &= A\hat{x}_k(t) + B\hat{u}_k(t) + \hat{w}_k(t) \\ \hat{y}_k(t) &= C\hat{x}_k(t) + \hat{v}_k(t) \end{aligned} \quad (19)$$

where $\Delta\bar{u}_k \triangleq \bar{u}_k - \bar{u}_{k-1}$ and $y_k(t) = \bar{y}_k(t) + \hat{y}_k(t)$.

4.2 ILC Formulation

The role of ILC is to compute $\bar{\mathbf{u}}_k$ using the information up to the $k-1^{\text{th}}$ run. For this, we first define

$$\begin{aligned} \bar{\mathbf{e}} &\triangleq \left[\, \bar{e}(1)^T \;\bar{e}(2)^T \;\cdots\; \bar{e}(N)^T \,\right]^T \\ \Delta\bar{\mathbf{u}} &\triangleq \left[\, \Delta\bar{u}(0)^T \;\Delta\bar{u}(1)^T \;\cdots\; \Delta\bar{u}(N-1)^T \,\right]^T \end{aligned} \quad (20)$$

where $\bar{e}(t) \triangleq y_d(t) - \bar{y}(t)$ and similarly for other variables. Through straightforward derivation from (18) and (19), we have

$$\begin{aligned} \bar{\mathbf{e}}_k &= \bar{\mathbf{e}}_{k-1} - \mathbf{G}\Delta\bar{\mathbf{u}}_k + \eta_k \\ \mathbf{e}_{k-1} + \mathbf{G}\hat{\mathbf{u}}_{k-1} &= \bar{\mathbf{e}}_{k-1} + \xi_{k-1} \end{aligned} \quad (21)$$

where

$$\mathbf{G} \triangleq \begin{bmatrix} CB & 0 & \cdots & 0 \\ CAB & CB & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ CA^{N-1}B & CA^{N-2}B & \cdots & CB \end{bmatrix} \quad (22)$$

η_k is expressed by a linear combination of $\bar{x}_k(0)$, $\mathbf{m}_k$, and $\mathbf{n}_k$, and thus represents a RWNR disturbance. Likewise, ξ_k is composed of $\hat{x}_k(0)$, $\hat{\mathbf{w}}_k$, and $\hat{\mathbf{v}}_k$, and also represents a RWNR disturbance.

Importing the idea of QILC (quadratic-criterion ILC) (Lee *et al.*, 2000), $\Delta\bar{\mathbf{u}}_k$ is determined to satisfy

$$\min_{\Delta\bar{\mathbf{u}}_k} \frac{1}{2}\left\{\|\bar{\mathbf{e}}_{k|k-1}\|^2_{\mathbf{Q}} + \|\Delta\bar{\mathbf{u}}_k\|^2_{\mathbf{S}}\right\} \quad (23)$$

where $\bar{\mathbf{e}}_{k|k-1}$ is the optimum prediction of $\bar{\mathbf{e}}_k$, which is given by the Kalman filter applied to (21). The unconstrained solution to (23) is

$$\Delta\bar{\mathbf{u}}_k = (\mathbf{G}^T\mathbf{Q}\mathbf{G} + \mathbf{S})^{-1}\mathbf{G}^T\mathbf{Q}\bar{\mathbf{e}}_{k-1|k-1} \quad (24)$$

where $\bar{\mathbf{e}}_{k-1|k-1}$ is obtained form the following Kalman filter equation:

$$\begin{aligned} \bar{\mathbf{e}}_{k-1|k-2} &= \bar{\mathbf{e}}_{k-2|k-2} - \mathbf{G}\Delta\bar{\mathbf{u}}_{k-1} \\ \bar{\mathbf{e}}_{k-1|k-1} &= \bar{\mathbf{e}}_{k-1|k-2} + \mathbf{K}(\mathbf{e}_{k-1} + \mathbf{G}\hat{\mathbf{u}}_{k-1} - \bar{\mathbf{e}}_{k-1|k-2}) \end{aligned} \quad (25)$$

4.3 RFC Formulation

RFC determines $\hat{u}_k(t)$ in real-time during the k^{th} run. For this, it is necessary to have a model that relates $\hat{u}_k(t)$ to $y_k(t)$. In order to construct the model, we first calculate the output by $\Delta\bar{u}_k(t)$ using the following relationship:

$$\begin{aligned} a_k(t+1) &= Aa_k(t) + B\Delta\bar{u}_k(t) \\ y_{a,k}(t) &= Ca_k(t) \end{aligned} \quad (26)$$

Subtracting (26) from (18) to eliminate $\Delta\bar{u}_k(t)$ yields

$$\begin{aligned} \bar{x}_{a,k}(t+1) &= A\bar{x}_{a,k}(t) + m_k(t) \\ \bar{y}_k(t) &= C\bar{x}_{a,k}(t) + y_{a,k}(t) + \bar{y}_{k-1}(t) + n_k(t) \end{aligned} \quad (27)$$

Combining (19) and (27) gives

$$\begin{aligned} \begin{bmatrix} \hat{x}_k(t+1) \\ \bar{x}_{a,k}(t+1) \end{bmatrix} &= \begin{bmatrix} A & 0 \\ 0 & A \end{bmatrix} \begin{bmatrix} \hat{x}_k(t) \\ \bar{x}_{a,k}(t) \end{bmatrix} \\ &\quad + \begin{bmatrix} B \\ 0 \end{bmatrix} \hat{u}_k(t) + \begin{bmatrix} \hat{w}_k(t) \\ m_k(t) \end{bmatrix} \quad (28) \\ y_k(t) - y_{a,k}(t) &= \left[\, C \;\, C \,\right] \begin{bmatrix} \hat{x}_k(t) \\ \bar{x}_{a,k}(t) \end{bmatrix} \\ &\quad + \bar{y}_{k-1}(t) + \hat{v}_k(t) + n_k(t) \end{aligned}$$

In the above, $\bar{y}_{k-1}(t)$ can be reasonably approximated by $\bar{y}_{k-1|k-1}(t)$, which is an optimal estimate of $\bar{y}_{k-1}(t)$ on the basis of the information up the k^{th} run. Then (28) can be written in a simplified form

$$\begin{aligned} z_k(t+1) &= \Phi z_k(t) + \Gamma\hat{u}_k(t) + \zeta_k(t) \\ y_k(t) - y_{a,k}(t) - \bar{y}_{k-1|k-1}(t) &= \Sigma z_k(t) + \nu_k(t) \end{aligned} \quad (29)$$

We consider the LQG criterion for $\hat{u}_k(t)$:

$$\min_{\hat{u}_k(\cdot)} E\left\{ \sum_{t=0}^{N-1} \|e_k(t)\|^2_Q + \|\hat{u}_k(t)\|^2_S \right\} \quad (30)$$

subject to (29)

Enforcing $e_k(t) \to 0$ is equivalent to steering $o_k(t) \triangleq y_k(t) - y_{a,k}(t) - \bar{y}_{k-1|k-1}(t)$ to $y_d(t) - y_{a,k}(t) - \bar{y}_{k-1|k-1}(t)$. Hence, (30) is a standard LQG servo problem for the output of (29) to follow $y_d(t) - y_{a,k}(t) - \bar{y}_{k-1|k-1}(t)$. The solution is standard and given in the following form:

$$\begin{aligned} \hat{u}_k(t) &= -L_{fb}(t)z_k(t|t) + L_{ff}(t)b_k(t+1) \quad (31) \\ \to \quad u_k(t) &= \bar{u}_k(t) + \hat{u}_k(t) \end{aligned}$$

where $z_k(t|t)$ represents the state estimate from the Kalman filter applied to (29). Detailed forms of $L_{fb}(t)$ and $L_{ff}(t)$ can be found from literature like (Lewis and Syrmos, 1995) or (Lee *et al.*, 2001).

One may handle $\bar{y}_{k-1}(t)$ in (28) in a rigorous manner instead of approximation. For such treatment, please refer to Lee *et. al*(2001).

4.4 Tuning Guideline

The proposed TBC technique has eight tuning factors: four from the covariance matrices of the noise terms in (18) and (19) for the Kalman filter design and the other four from the weighting factors of the quadratic criteria in (23) and (30). Among them, the weighting factor tuning is rather transparent whereas the covariance matrix tuning needs some discussion. Since the most crucial thing to the proposed control technique is how to correctly estimate $\bar{\mathbf{e}}_k$ by the associated Kalman filters, we give brief tuning guidelines for the covariance matrices using R to denote the covariance matrices.

We start from the ILC part. From the nature of the Kalman filter for (21), a large R_η to R_ξ ratio results in weak filtering of RWNR disturbances (much confidence on the measurement values). Such a choice should be made when the real-time disturbance is not large or a newly entered disturbance is supposed to last for ensuing runs. In the opposite case, opposite choice of the covariance matrices is needed. Here, ξ_k is composed of m_k and n_k, and η_k of $\hat{w}_k$ and $\hat{v}_k$. Hence, increasing R_ξ can be made by increasing $R_{\hat{w}}$ and/or $R_{\hat{v}}$ and likewise for R_η. Now, from (28) we can see that the four covariance matrices also determine the behavior of the RFC Kalman filter. In this case, however, $\hat{v}_k$ and n_k act as the measurement noise, and $\hat{w}_k$ and m_k as the process noise. In most cases, especially in most chemical engineering problems, the process noise dominates the measurement noise, hence R_m and $R_{\hat{w}}$ practically determine R_η and R_ξ, respectively.

On the basis of the above consideration, the tuning procedure can be recommended as follows: First, tune the ILC part covariance matrices R_η and R_ξ using R_m and $R_{\hat{w}}$, respectively. Next, tune R_n and $R_{\hat{v}}$ in relation to R_m and $R_{\hat{w}}$ for desired RFC performance.

5. NUMERICAL ILLUSTRATION

5.1 Linear SISO System

The plant and nominal models are the sampled-data versions of

$$\mathbf{G}^p(s) = \frac{2.5}{300s^2 + 35s + 1} \quad \text{and}$$
$$\mathbf{G}^m(s) = \frac{1.5}{270s^2 + 33s + 1}. \qquad (32)$$

with a sampling period of 1, respectively, over N =100 sampling steps. On the plant output, zero-mean measurement noise of variance 0.02^2 is added. The controller was designed using a state space conversion of the nominal model. It is assumed that a unit step signal filtered by $1.5/(10s+1)$ enters at the plant output as a disturbance from $t = 31$ to the batch terminal time. We considered two disturbance scenarios. In the first scenario, the disturbance occurs only at the 11^{th} run. In the second scenario, the same disturbance is repeated from the 11^{th} to 20^{th} runs.

Nominal values of tuning parameters were given as follows:

$$R_{\hat{w}}{=}0.1^2 I,\ R_{\hat{v}}{=}0.02^2 I,\ R_m{=}0.1^2 I,\ R_n{=}0.02^2 I,$$
$$\mathbf{Q} = I, \quad \mathbf{S} = 0.01I, \quad Q = I, \quad S = 0.01I \qquad (33)$$

5.2 Results and Discussion

In Fig. 1, the performance of the proposed control technique is shown for the first disturbance scenario. Through ten consecutive runs, the input as well as the output signals converge to their respective limits. When there enters a disturbance

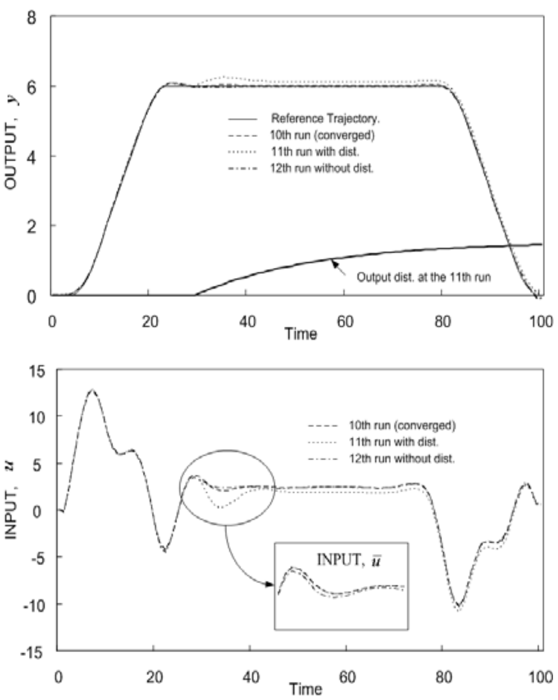

Fig. 1. Performance of the proposed batch control technique against a run-wise nonrecurrent disturbance.

at the 11^{th} run, its effect is almost completely isolated the 11^{th} run and virtually not carried to the 12^{th} run. The performance can be more lucidly observed from the learning input signal given in Fig. 1(b). We can see that the ILC signal $\bar{u}_k(t)$ doesn't change much by the RWNR disturbance while $\hat{u}_k(t)$ moves aggressively by the RFC action.

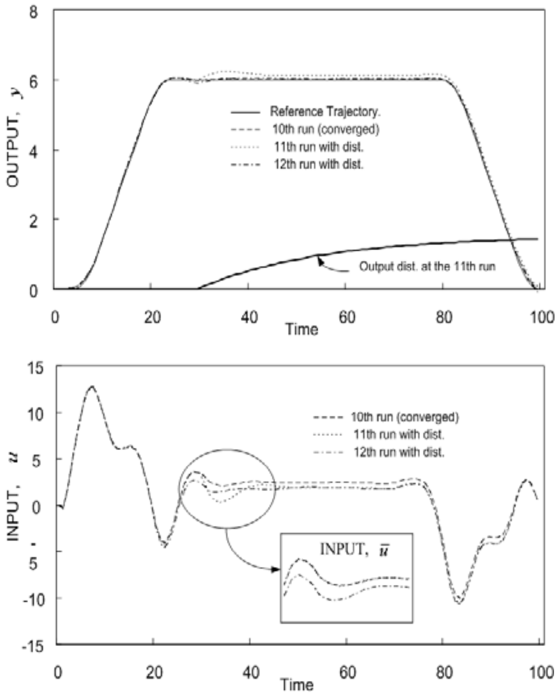

Fig. 2. Performance of the proposed batch control technique against a run-wise recurrent disturbance.

In Fig. 2, the performance for the second disturbance scenario is shown. In this scenario, the disturbance is repeated from the 11^{th} run and it is completely rejected by the learning control action as the run number increases. At the 11^{th} run, the controller regards the disturbance as a RWNR one and hence large $\hat{u}$ is exerted. However, as the same disturbance is repeated, $\bar{u}$ begins to change and eventually converges to a new profile that can compensate for the disturbance just as in ordinary ILC methods.

6. CONCLUSIONS

The two-stage batch control technique, where ILC and RFC can independently respond to the RWR and RWNR disturbances, respectively, has been developed based on the stochastic state space model. The key step to this derivation is the appropriate decomposition of the disturbance signal and input signal, and accompanied decomposition of the state space model. Numerical study reveals that the proposed technique works as anticipated overcoming the problems of existing RFC-ILC methods. It is believed that the advancement made by the present study will improve the practical applicability of ILC methods.

ACKNOWLEDGEMENTS

The first author would like to acknowledge the financial support from Korea Science and Engineering Foundation (KOSEF) through the Post-doctoral Fellowship Program. The last author (KSL) is grateful to the KOSEF for the financial support. The authors are grateful to the members of Texas-Wisconsin Modeling and Control Consortium (TWMCC) for their financial support.

REFERENCES

Chin, I., S. J. Qin, K. S. Lee, and M. Cho (2004). A two-stage iterative learning control techynique combined with real-time feedback for independent disturbance rejection. *Automatica*, in press.

Hashimoto, H and J.-X. Xu (1987). Learning control system with feedback. *Proc. of IEEE Asian Electronics Conference*, Hong Kong, 490–494.

Lee, J., K. S. Lee, and W. C. Kim (2000). Model-based iterative learning control with a quadratic criterion for time-varying linear systems. *Automatica* **36**, 641–657.

Lee, K. S., J. Lee, I. S. Chin, J. Choi, and J. H. Lee (2001). Control of wafer temperature uniformity in rapid thermal processing using an optimal iterative learning control technique. *Ind. & Eng. Chem. Res.*, **40**, 1661.

Lewis, F. L. and V. L. Syrmos (1995). *Optimal Control*, John Wiley and Sons, N.Y..

Moore, K. L. (1999). Iterative learning control - an expository overview. *Applied & Computational Controls, Signal Processing, and Circuits* **1**(1), 151–241.

Phan, M. Q., R. W. Longman, and K. L. Moore (2000). Unified formulation of linear iterative learning control. *Proc. of AAS/AIAA Space Flight Mechanics Meeting*, Clearwater, FL, Paper No. AAS00-106.

Xu, J.-X., X. Wang, and T. H. Lee (1995). Analysis of iterative learning control systems using current cycle feedback. *Proc. of ACC*, Seattle, USA, 1–8.