Proceedings of the 17th World Congress
The International Federation of Automatic Control
Seoul, Korea, July 6-11, 2008

Optimal Iterative Learning Control for Batch Processes with Model Parameter Variations Using Strong Tracking Filter

Yixin Xu , Zhihua Xiong, Yongheng Jiang, Dexian Huang

Department of Automation, Tsinghua University, Beijing 100084, P. R. China
(*e-mail: zhxiong@tsinghua.edu.cn*)

Abstract: An optimal iterative learning control (ILC) strategy is proposed to track product quality trajectories of batch processes by updating a linear time-varying perturbation (LTVP) model. To address the problem of model parameter variations from batch to batch, the LTVP model is renewed by using strong tracking filter (STF) algorithm. Comparing recursive least squares (RLS), STF can capture the changing dynamics of the process more accurately. The tracking error transition models can be built, and the ILC law with direct error feedback is explicitly obtained. Sufficient conditions of convergence are derived for the optimal ILC based on the LTVP model. It has also been proved that the tracking error will converge to a small constant but depend on the accuracy of the LTVP model error. If there is no model error, the tracking error can converge to zero. By using STF to update the LTVP model, the model accuracy is improved and the tracking control performance is also enhanced. The proposed strategy is illustrated on a typical batch reactor, and the results demonstrate that the performance of tracking product qualities can be improved under the proposed strategy when model parameter variations occur with respect to the batch index.

1. INTRODUCTION

The repetitive nature of batch process operations allows that the information of previous batch runs can be used to improve the operation of the next batch. Recently, iterative learning control (ILC) has been used in the batch-to-batch control of batch processes to directly update input trajectory (Lee and Lee, 2007). Refinement of control signals based on ILC can significantly enhance the performance of tracking control systems. Bristow *et al*. (2006) presents a survey of the major results based on linear models in ILC analysis and design over the past two decades.

Since ILC is well developed for linear models, most of the ILC-based batch-to-batch control schemes are based on some kinds of linear models, for instance, linear time-invariant system (Saab, 1995). Optimal ILC is one of important methods for designing an iterative learning law, in which the ILC law is derived from a quadratic objective function (Owens and Hatonen, 2005). Amann et al. (1996) proposes an optimal ILC based on optimization principle by combining the Riccati feedback control with the typical ILC feed-forward control. Shi et al. (2006) studied the optimal ILC further and proposed a general design framework for ILC of an injection mold process based on a two-dimensional (2D) system.

Lee and co-workers in several related articles (Lee and Lee, 1997; Lee et al., 2000; Lee and Lee, 2003) proposed the quadratic criterion-based ILC (Q-ILC) approach for tracking control of batch processes. Lee et al. (2000) combines the advantages of ILC and MPC into a single framework. A batch MPC (BMPC) technique and its extension for tracking control are proposed by incorporating the capability of real-time feedback control into quadratic criterion-based ILC (Q-ILC). The proposed approach is applied to tracking control for temperature of batch processes based on a linear time-varying (LTV) tracking error transition model.

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. Therefore, the convergence of iterative learning law is an important issue in the design and application of ILC. In our previous work (Xiong and Zhang, 2003; Xiong et al, 2005), an ILC strategy for the tracking control of product quality in batch processes is proposed based on an LTVP model. In practice, the LTVP model of product quality can be obtained by linearizing nonlinear model with respect to the nominal trajectories.

However, in some cases that model parameters change with respect to batch index, previous control policy and learning algorithm may not as useful as previous experience has become invalid but the experience is still used in new batches. For example, in a chemical reactor, the process parameter may change along with time or iteration number because of those uncertainties in reactor conditions, such as impurities, raw material conditions and so on (Xiong et al, 2005). This is probably dangerous in some particular time for the reason that the model parameter change may result in instability. In other cases, this may slow the learning speed. When model uncertainties occur in such way, the LTVP model can be updated from batch to batch by using strong tracking filter (STF) algorithm (Zhou and Frank,1996; Wang et al., 2004), which can capture the renewed dynamics of the process when process variations exist. STF has strong tracking ability to the model no matter whether the model parameters change abruptly or slowly, and whether the process has reached steady state or not, and it has definite robustness against model uncertainties (Zhou and Frank,1996).

The optimal ILC based on updated LTVP model is proposed to address this problem due to model parameter variations, which is quite similar to the adaptive ILC (Owens. and Munde, 1998; Tayebi and Chien, 2006). Despite of system parameter variation, the ILC algorithm has the ability to ensure that the system is still stable and as well has a fast learning speed. Similar to our previous work, a nonlinear model considered is converted to an LTVP model. As the same as optimal ILC, a cost function is minimized to detain control profile. Parameters of the optimal ILC law will vary with the estimated LTVP model while batch index increases. Because of these features of the learning law, the algorithm can cope with nonlinear systems as well system parameters change.

The rest of this paper is organized as follows: Section 2 presents the problem solved by this paper. In Section 3, the approach based on LTVP model is proposed and convergence of the algorithm is also proved. In section 4, strong tracking filter is proposed to estimate the LTVP model. In Section 5, simulation results are given to demonstrate the proposed method. Finally Section 6 draws some concluding remarks.

2. PPOBLEM FORMULATION

In this study, the batch processes considered is written by (Xiong, et al., 2005)

$$\begin{aligned} \dot{x}_k(t) &= f(\, x_k(t),\ u_k(t)\,) \\ y_k(t) &= g(\, x_k(t),\ u_k(t)\,) \end{aligned} \tag{1}$$

where $x \in R^n$ is system state, $u \in R^m$ the input, $y \in R^q$ the output, $t \in [0, T_f]$ the time interval, subscript k the batch index, $f: \mathrm{R}^n \times \mathrm{R}^m \to \mathrm{R}^n$, $g: \mathrm{R}^n \times \mathrm{R}^m \to \mathrm{R}^q$, f and g nonlinear function, respectively.

To relate control profile to output profile over the whole batch duration, a batch-wise nonlinear function can be derived as

$$y_k(t) = F(\ u_k(t)\,) \tag{2}$$

where $F: \mathrm{R}^m \to \mathrm{R}^q$. The time interval consists of N_f sampling times, and each sampling time $h = T_f/N_f$ is equal. Let us define

$$\begin{aligned} U_k &= [u_k(0) \quad u_k(1) \quad \cdots \quad u_k(N_f-1)]^T \\ Y_k &= [y_k(1) \quad y_k(2) \quad \cdots \quad y_k(N_f)]^T \end{aligned} \tag{3}$$

The object of ILC is to find an optimal strategy of control policy U_k to make sure that system output Y_k is in some neighborhood of expected output Y_d despite of system parameters change when batch index increases, that is,

$$\lim_{k\to\infty} \left\| Y_d - Y_k \right\| \le \sigma < \infty \tag{4}$$

where $Y_d = [y_d(1) \quad y_d(2) \quad \cdots \quad y_d(N_f)]^T$.

3. OPTIMAL ITERATIVE LEARNING CONTROL

3.1 Derivation of the Optimal ILC Law

Model (2) considered above can be rewritten as

$$\begin{aligned} y_k(1) &= f_1(u_k(0)) \\ y_k(2) &= f_2(u_k(0), u_k(1)) \\ &\vdots \\ y_k(N_f) &= f_{N_f}(u_k(0), u_k(1), \cdots, u_k(N_f-1)) \end{aligned} \tag{5}$$

Let us define

$$U_k(t) = [u_k(0) \quad u_k(1) \quad \cdots \quad u_k(t-1)]^T \tag{6}$$

where $t = 1, 2, \cdots, N_f$, then above (5) can be written as

$$y_k(t) = f_t(U_k(t)) \tag{7}$$

Taylor series expansion of model (7) with respect to the batch is calculated as follows (Xiong and Zhang, 2003)

$$y_{k+1}(t) = y_k(t) + \left. \frac{\partial f_t(U_{k+1}(t))}{\partial U_{k+1}(t)} \right|_{U_k} (U_{k+1}(t) - U_k(t)) + w_{k+1}(t) \tag{8}$$

where $w_{k+1}(t)$ is high order of Taylor series which is called as model error.

Let us define the following perturbation variables

$$\Delta y_{k+1}(t) = y_{k+1}(t) - y_k(t) \tag{9}$$

$$\Delta U_{k+1}(t) = U_{k+1}(t) - U_k(t) \tag{10}$$

$$G_{k+1}(t) = \left. \frac{\partial f_t(U_{k+1}(t))}{\partial U_{k+1}(t)} \right|_{U_k} \tag{11}$$

Then the perturbation model of linearized form is derived as

$$\Delta y_{k+1}(t) = G_{k+1}(t) \Delta U_{k+1}(t) + w_{k+1}(t) \tag{12}$$

Rewriting all equations in (5) in the form of matrix, then we have

$$\Delta Y_{k+1} = G_{k+1} \Delta U_{k+1} + w_{k+1} \tag{13}$$

Considering the expected output Y_d, output tracking error of the (k+1)th batch is defined as

$$e_{k+1} = Y_d - Y_{k+1} \tag{14}$$

The object of optimal ILC is to find input U_{k+1} to minimize the following cost function

$$J_{k+1}(\Delta U_{k+1}) = \Delta U_{k+1}^T R \Delta U_{k+1} + e_{k+1}^T Q e_{k+1} \tag{15}$$

where Q and R are definite symmetric matrices. By minimizing equation (15) and through straightforward manipulation, the following optimal ILC law is obtained

$$U_{k+1} = U_k + K_{k+1} e_k \tag{16}$$

where K_{k+1} is the learning gain, and

$$K_{k+1} = \left[G_{k+1}^T Q G_{k+1} + R\right]^{-1} G_{k+1}^T Q \tag{17}$$

It can be seen from (16)(17) that the learning gain K_{k+1} of the optimal ILC varies from the change according to the model parameter G_{k+1} and if model parameters change, K_{k+1} will also change, then the optimal ILC learning law is obtained.

3.2 Convergence Analysis

Theorem: Let $\{e_{k+1}\}$ be the tracking error sequence under the above optimal ILC law (16). If the following conditions are satisfied, $\forall k = 1,2,\cdots,\infty$

$$\|I - G_k K_k\| \le \varepsilon < 1 \tag{18}$$

$$\|w_{k+1}\| \le E < \infty \tag{19}$$

then system output will converge to a neighborhood of expected output, and output tracking error can converge to a small value.

Proof: Considering (16), we have,

$$\Delta U_{k+1} = K_{k+1} e_k \tag{20}$$

Substituting (20) to the LTVP model (13), we can obtain

$$\Delta Y_{k+1} = G_{k+1}\Delta U_{k+1} + w_{k+1} = G_{k+1}K_{k+1}e_k + w_{k+1} \tag{21}$$

And we also have

$$\Delta Y_{k+1} = Y_{k+1} - Y_k = (Y_d - e_{k+1}) - (Y_d - e_k) = e_k - e_{k+1} \tag{22}$$

Thus iteration relationship of tracking error can be obtained

$$e_{k+1} = e_k - G_{k+1}K_{k+1}e_k - w_{k+1} \tag{23}$$

If the above relationship of tracking error is calculated recursively, we have

$$\begin{aligned}
&e_{k+1} \\
&= e_k - G_{k+1}K_{k+1}e_k - w_{k+1} \\
&= (I - G_{k+1}K_{k+1})e_k - w_{k+1} \\
&= (I - G_{k+1}K_{k+1})(I - G_k K_k)e_{k-1} - (I - G_{k+1}K_{k+1})w_k - w_{k+1} \\
&= \cdots \\
&= \left[\prod_{i=2}^{k+1}(I - G_i K_i)\right]e_1 - \sum_{i=2}^{k+1}\left[w_i \prod_{j=1+i}^{k+1}(I - G_j K_j)\right]
\end{aligned} \tag{24}$$

where e_1 is the initial tracking error and is assumed to be obtained. It can be found further that

$$\begin{aligned}
&\|e_{k+1}\| \\
&= \left\| \left[\prod_{i=2}^{k+1}(I - G_i K_i)\right]e_1 - \sum_{i=2}^{k+1}\left[w_i \prod_{j=1+i}^{k+1}(I - G_j K_j)\right] \right\| \\
&\le \|e_1\| \prod_{i=2}^{k+1}\|I - G_i K_i\| + \left\| \sum_{i=2}^{k+1}\left[w_i \prod_{j=1+i}^{k+1}(I - G_j K_j)\right] \right\|
\end{aligned} \tag{25}$$

Considering the conditions (18), we have

$$\|e_{k+1}\| \le \|e_1\|\varepsilon^{k-1} + \left\| \sum_{i=2}^{k+1}\left[w_i \prod_{j=1+i}^{k+1}(I - G_j K_j)\right] \right\| \tag{26}$$

Considering the condition (19) which is directly related to system model error w_{k+1} in (13), we have

$$\begin{aligned}
&\left\| \sum_{i=2}^{k+1}\left[w_i \prod_{j=1+i}^{k+1}(I - G_j K_j)\right] \right\| \le \sum_{i=2}^{k+1}\left(\|w_i\| \prod_{j=1+i}^{k+1}\|I - G_j K_j\| \right) \\
&\le E \sum_{i=2}^{k+1}\left(\prod_{j=1+i}^{k+1}\varepsilon \right) = \frac{E(1-\varepsilon^k)}{1-\varepsilon}
\end{aligned} \tag{27}$$

Substituting (27) to (26), we can find

$$\|e_{k+1}\| \le \|e_1\|\varepsilon^{k-1} + \frac{E(1-\varepsilon^k)}{1-\varepsilon} \tag{28}$$

Then the following convergence of e_{k+1} can be obtained

$$\lim_{k\to\infty}\|Y_d - Y_{k+1}\| = \lim_{k\to\infty}\|e_{k+1}\| \le \frac{E}{1-\varepsilon} = \eta \tag{29}$$

It means that the tracking error e_{k+1} converges to a small value with respect to the batch index.

□

From analysis above, we can derive that the accuracy of LTVP model has large contribution to the tracking performance of system output. The smaller the system model error, the smaller the output tracking error. In particular, if there is no model error, the tracking error can converge to zero.

4. MODEL PRARAMETERS IDENTIFICATION USING STRONG TRACKING FILTER

In this section, strong tracking filter (STF) (Zhou and Frank, 1996) is used to solve the problem when system model parameters change with respect to the batch number. STF has strong tracking ability to the model no matter whether the model parameters change abruptly or slowly, and whether the process has reached steady state or not, and it has definite robustness against model uncertainties.

The model to use STF is

$$\Delta y_{k+1}(t) = \Delta U_{k+1}(t)^T G_{k+1}(t)^T + w_{k+1}(t) \tag{30}$$

And the algorithm to identify $G_{k+1}(t)^T$ is

$$\hat{G}_{k+1|k+1}(t)^T = \hat{G}_{k+1|k}(t)^T + S(k+1)\gamma(k+1) \tag{31}$$

$$\hat{G}_{k+1|k}(t)^T = \hat{G}_{k|k}(t)^T \tag{32}$$

where

$$\begin{aligned}
S(k+1) = {} & P(k+1\,|\,k)H^T(k+1)\cdot \\
& [H(k+1)P(k+1\,|\,k)H^T(k+1) + R(k)]^{-1}
\end{aligned} \tag{33}$$

$$P(k+1\,|\,k) = L(k+1)P(k\,|\,k) \tag{34}$$

$$P(k+1\,|\,k+1) = [I - S(k+1)H(k+1)]P(k+1\,|\,k) \tag{35}$$

$$\gamma(k+1) = \Delta y_{k+1}(t) - \Delta U_{k+1}(t)^T \hat{G}_{k+1|k}(t)^T \tag{36}$$

where

$$H(k+1) = \Delta U_{k+1}(t)^T \tag{37}$$

$$L(k+1) = diag\{\lambda_1(k+1), \lambda_2(k+1), \cdots, \lambda_t(k+1)\} \tag{38}$$

$$\lambda_i(k+1) = \begin{cases} \alpha_i \eta(k+1), & \alpha_i \eta(k+1) > 1 \\ 1 & \alpha_i \eta(k+1) \le 1 \end{cases} \tag{39}$$

$$\eta(k+1) = \frac{\mathrm{tr}[\mathbf{N}(k+1)]}{\sum_{i=1}^{n} \alpha_i \mathbf{M}_{ii}(k+1)} \tag{40}$$

$$N(k+1) = V_0(k+1) - \beta \cdot R(k+1) \tag{41}$$

$$M(k+1) = P(k|k) \cdot H^T(k+1)H(k+1) = (M_{ij}) \tag{42}$$

$$V_0(k+1) = E[\gamma(k+1)\gamma^T(k+1)] \approx \begin{cases} \gamma(1)\gamma^T(1), k=0 \\ \dfrac{\left[\rho V_0(k) + \gamma(k+1)\gamma^T(k+1)\right]}{1+\rho}, k \ge 1 \end{cases} \tag{43}$$

where ρ is the forgetting factor, $\beta \ge 1$ is weaken factor and determined reasonably, $\alpha_i \ge 1$ $(i = 1,2,\cdots,t)$ are coefficients.

As we can see from (31)-(36) that, in some point of view, the STF algorithm is improved from the RLS algorithm. The main difference between these two algorithms is that the term $L(k+1)$ in (34) is used. It is a key to the STF algorithm. In RLS, we often find that the matrix P in (34) converges to zero as the batch number increases, which is almost right if model parameters do not change. As a result, the model parameter G_{k+1} converges to its real value. However, as model parameters change in some cases, for instance in our application to a typical batch reactor, it will cause that in (36), the term $\gamma(k+1)$ changes along with real model parameter. Because matrix P is almost zero, and then is $S(k+1)$ in (33) will be also zero, which causes in further steps that the estimated model parameter G_{k+1} in (31) changes very little despite of real model parameter changes in a large scale. Considering the previous theorem in section 3.2, we can see that the model error will increase a lot and then the tracking error increases. It is for this reason that we proposed the STF algorithm in order to improve the model error, thus the tracking error is decreased compared with the RLS algorithm. As stated in Zhou and Frank (1996) and Wang et al. (2004), the matrix $L(k+1)$ can be obtained through minimization of cost function. For details, see Zhou and Frank (1996).

5. APPLICATION TO A TYPICAL BATCH REACTOR

Consider a typical nonlinear batch reactor with temperature as the control variable (Logsdon and Biegler, 1989). The reaction scheme is $A \xrightarrow{k_1} B \xrightarrow{k_2} C$, and the operation objective of the reactor is to maximize the product (B) after a fixed time. The equations describing the batch process are

$$\frac{dx_1}{dt} = -k_1 \exp(-E_1/uT_{ref})x_1^2 \tag{44}$$

$$\frac{dx_2}{dt} = k_1 \exp(-E_1/uT_{ref})x_1^2 - k_2 \exp(-E_2/uT_{ref})x_2 \tag{45}$$

where x_1 and x_2 are the dimensionless concentrations of product A and B, $u=T/T_{ref}$ is the dimensionless temperature of the reactor and T_{ref} is the reference temperature, respectively. The parameters are set to be k_1=4.0×10^3, k_2=6.2×10^5, E_1= 2.5×10^3, E_2=5.0×10^3 and T_{ref}=348. The final time t_f is 1.0h, initial conditions are $x_1(0)=1$ and $x_2(0)=0$, and the reactor temperature is constrained to 298 K ≤ T ≤ 398 K (i.e. 0.856≤ u ≤1.144). Here the batch length is divided into N_f=10 equal stages.

In this study, the above mechanistic models (44) and (45) are assumed not to be available. Because the objective of the reactor is to maximize the product (B), an LTVP model is built to model the relationship between $y=x_2$ and u. The desired product reference trajectory Y_d was selected from Logsdon and Biegler (1989). The parameters of ILC were set as $Q=\boldsymbol{I}$ and $R=0.02\boldsymbol{I}$, and the parameters in STF were set as $\rho = 0.95$, $\beta = 1$ and $\alpha_i = 1$ $(i = 1,2,\cdots,t)$.

Here the model parameter changes are simulated by batch-to-batch parametric change. The scenario here is that the kinetic parameter k_1 increases by 100 at the 10th batch and then keeps constant, E_2 increases by 250 at the 10th batch, as shown in Fig 1.

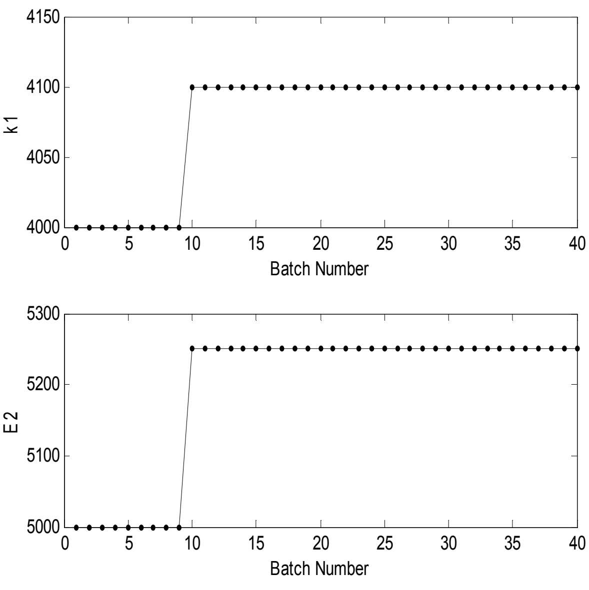

Fig. 1. Model parameter variations of k_1 and E_2

To compare the performance of STF, the recursive least square (RLS) method is also used to estimate the LTVP model. Fig 2 shows the results that the output tracking errors converge along the batch index under the proposed optimal ILC using both STF and RLS to identify the model. It can be found from Fig 2 that when there is no model parameter variation before 10 batches, the output tracking error can converge quickly under the proposed optimal ILC while the perturbation model is gradually identified by STF and RLS.

When model parameter variations occur after the 10th batch run, the tracking errors are under a small value with the optimal ILC due to the updated model and learning rate. After 30 batch runs, the tracking errors in the two cases converge to small values when the model parameters do not change. But the LTVP model is estimated more accurately by STF, the tracking error converges to a smaller value than the value in the case using RLS. It is also demonstrated that the tracking control performance of the optimal ILC is based on the model accuracy in the learning law.

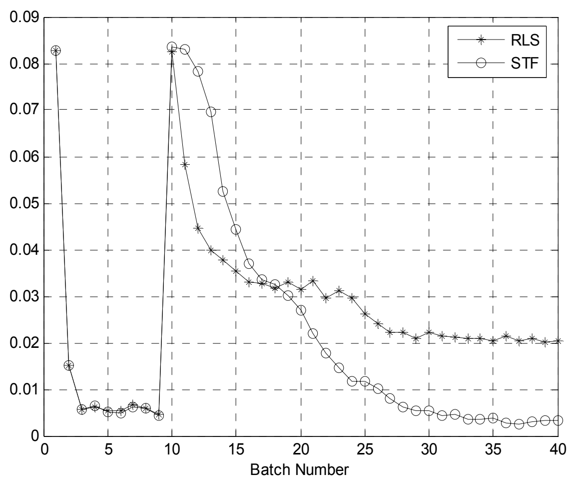

Fig. 2. Output tracking error convergence under the optimal ILC based on LTVP model updated by STF and RLS

Fig 3 and Fig 5 show that the output trajectories converge gradually with respect to the batch number under the proposed optimal ILC when using RLS and STF to estimate the LTVP model. To figure out the difference clearly, Fig 4 and Fig 6 show the output profiles at time t=0.6~1.0h. It can be seen that at the 10th batch run, when there are model parameter changes, the output profiles are far away from the expected output trajectory due to model mismatches in both RLS case and STF case. After updating the model by RLS and STF, the output profiles approach to the expected one gradually. Due to the more accurate model by STF than that by RLS, the performance of ILC in STF case is improved a little, especially there are less tracking errors after 30 batch runs.

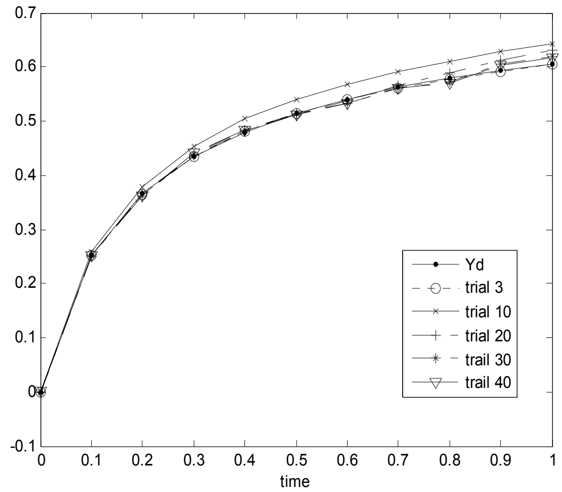

Fig. 3. Output trajectory convergence under the optimal ILC based on updated LTVP model by RLS

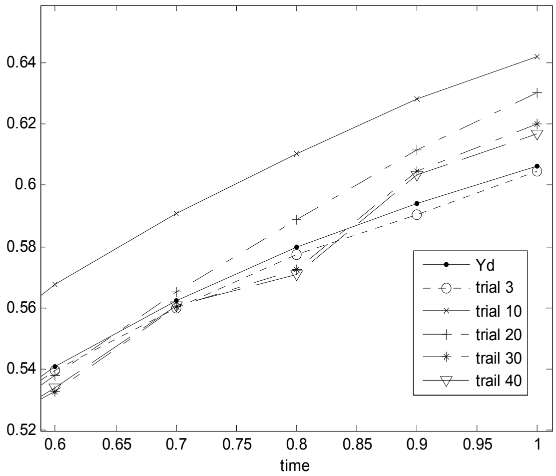

Fig. 4. Output trajectory convergence at time t=0.6~1.0h

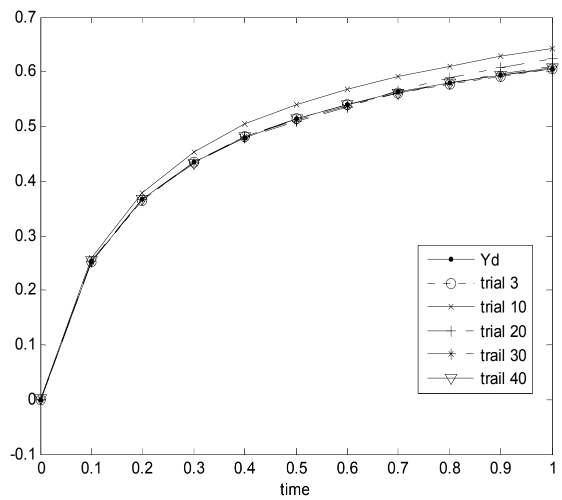

Fig. 5. Output trajectory convergence under the optimal ILC based on updated LTVP model by STF

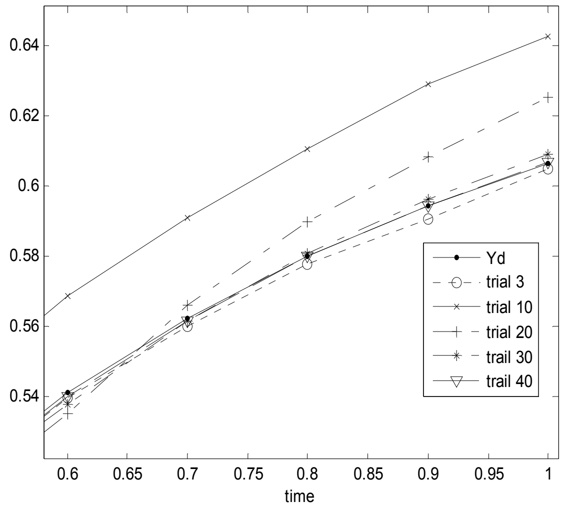

Fig. 6. Output trajectory convergence at time t=0.6~1.0h

6. CONCLUSIONS

An optimal iterative learning control algorithm based on LTVP model is proposed to address the problem of model parameter variations. To address the problem of model parameter variations from batch to batch, the LTVP model is updated by using strong tracking filter (STF) algorithm. Conditions of convergence are also derived for the proposed optimal ILC. It has been proved that the tracking error can converge to a small constant value. Simulation results on a typical batch reactor have shown that the output tracking error converges while the model parameters vary with respect to the batch index.

ACKNOWLEGEMENTS

The authors thank the financial support of National Natural Science Foundation of China (60404012, 60674064, 60704032), National High-Tech 863 Project (No. 2007AA-04Z193), New Star of Science and Technology of Beijing City (2006A62), and IBM China Research Lab 2007 UR-Program.

REFERENCES

Amann, N., D.H. Owens, E. Rogers (1996) Iterative learning control for discrete-time system with exponential rate of convergence. *IEE Proc. D -Control Theory Applications*, **143**, 217-224.

Bristow, D.A., M. Tharayil and A.G. Alleyne (2006) A survey of iterative learning control. *IEEE Control Systems Magazine* , **26**, 96-114.

Lee, J.H., K.S. Lee (2007) Iterative learning control applied to batch processes: An overview. *Control Engineering Practice*, in press, available online.

Lee, J.H., K.S. Lee, 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. and J.H. Lee (1997) Model predictive control for nonlinear batch processes with asymptotically perfect tracking. *Computers Chem. Engng.*, **21**, s873-s879.

Lee, K.S. and J.H. Lee (2003) 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*, **13**, 607-621.

Logsdon, J. S. and L.T. Biegler (1989). Accurate solution of differential-algebraic optimization problems. *Ind. Eng. Chem. Res.*, **28**, 1628-1639.

Owens, D.H. and J. Hatonen (2005) Iterative learning control – An optimization paradigm. *Annual Reviews in Control*, **29**, 57-70.

Owens, D.H. and G.S. Munde (1998). Universal adaptive iterative learning control. In: *Proceedings of the 37th IEEE Conference on Decision and Control*, vol **1**, 181-185.

Saab, S. A. (1995) Discrete-time learning control algorithm for a class of linear time-invariant systems. *IEEE Trans. Automat. Control*, **40**, 1138-1142.

Shi, J., F.R. Gao, T.J. Wu (2006) Robust iterative learning control design for batch processes with uncertain perturbations and initialization. *AIChE J.*, **52**, 2171-2187.

Tayebi, A. and Chiang-Ju Chien (2006). An adaptive iterative learning control framework for a class of uncertain nonlinear systems. In: *Proceedings of the 45th IEEE Conference on Decision and Control*, 5054 – 5059

Wang, D., D.H. Zhou, Y.H. Jin, S. Joe Qin (2004) A strong tracking predictor for nonlinear processes with input time delay. *Computers and Chemical Engineering* **28**, 2523–2540

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

Xiong, Z.H., J. Zhang, X. Wang, Y.M. Xu (2005). Tracking control for batch processes by integrating batch-to-batch iterative learning control and within-batch online control. *Ind. Eng. Chem. Res.*, **44**, 3983-3992.

Zhou, D.H. and P.M. Frank (1996). Strong tracking filtering of nonlinear time-varying stochastic systems with colored noise: Application to parameter estimation and empirical robustness analysis. *International Journal of Control*, **65**(2), 295–307