Preprints of the 10th IFAC International Symposium on Dynamics and Control of Process Systems
The International Federation of Automatic Control
December 18-20, 2013. Mumbai, India

Two dimensional recursive least squares for batch processes system identification

Zhixing Cao * **Yi Yang** ** **Jingyi Lu** *** **Furong Gao** ****

* *Dept. of Chemical and Biomolecular Engineering, Hong Kong University of Science and Technology, Clear Water Bay, Kowloon, Hong Kong(e-mail: zcaoab@ust.hk).*
** *Dept. of Control Science and Engineering, Zhejiang University, Hangzhou, China(e-mail: yangyi@iipc.zju.edu.cn)*
*** *Dept. of Chemical and Biomolecular Engineering, Hong Kong University of Science and Technology, Clear Water Bay, Kowloon, Hong Kong (e-mail: jluab@ust.hk)*
**** *Dept. of Chemical and Biomolecular Engineering, Hong Kong University of Science and Technology, Clear Water Bay, Kowloon, Hong Kong (e-mail: kefgao@ust.hk)*

Abstract: Recursive system identification is an important problem in many advanced control techniques, such as adaptive control. This paper presents a new approach of two dimensional recursive least squares identification method suitable for batch processes. In this way, system identification is carried out not only using the information from time direction within the batch but also from batch to batch direction. A constraint term is incorporated in the cost function to reduce parameters varying. A guideline for selecting weight matrix in application is also provided. Furthermore, simulation results based on the data obtained from a model of injection moulding, a typical batch process, are illustrated to testify the superiority of the proposed method over the conventional recursive leasts squares.

Keywords: Two dimensional, system identification, recursive least squares(RLS)

1. INTRODUCTION

Batch process is getting more and more important nowadays in industries, due to its capability of manufacturing high-value-added products with superior versatility. The application of batch process includes, for example, injection moulding, semi-conduct, pharmaceutical industry, etc. Controller design is a pivotal problem in batch processes to guarantee the final quality of products. Many advanced control strategies have been proposed to improve the control performances of batch process, in order to ensure product quality. Many of them share a common feature that the performance of control strategies highly depends on the precision and reliability of process model. How to obtain a suitable process model is a critical issue. Owing to the nature of batch process, i.e. highly nonlinearity, complex mechanism small production volume, and rapid changing market conditions, a black-box approach, or called system identification, is an appropriate and desirable way to achieve that goal rather than a white-box approach which is based on the first principles. System identification has been studied extensively for decades by many researchers, e.g., Ljung (2009), Söderström and Stoica (1989), Eykhoff and Pieter (1974) and references therein, for application Zhu (2001), Mehra and Lainiotis (1976) and references therein.

System identification is widely adopted for implementing for the rapid development of modern computer system. Quite a few successful applications to batch processes have been reported. Based on RLS and second-order autoregressive exogenous model (ARX) structure, Yang and Gao (1998, 1999) applied self tuning control with pole placement and generalised predictive control (GPC) to both nozzle packing pressure and injection velocity on injection moulding machine successfully with supreme performances. Shi et al. (2005) designed a robust iterative learning control integrated with feedback control achieving a very good result on controlling injection velocity based on an identified second-order ARX model. Shi et al. (2007) further proposed a two dimensional GPC controller, and also tested on an identified ARX model with an excellent outcome.

Nevertheless, all the methods mentioned above were simply borrowing system identification approaches designed for continuous processes ignoring the properties of batch processes. There are very few publications on this aspect, to the best of the authors knowledge. Ma and Braatz (2003) developed an iterative way to identify a model for batch processes and minimise the model uncertainty in the mean time. However, all of these procedures were conducted off-line and involves large computation. Tayebi (2004) developed a two dimensional approach for some unknown parameters of robot manipulator in a continuous type, but did not provide results on the performance of the estimator. Chi et al. (2008) designed a discrete-time two dimensional system identification method integrated with adaptive iterative learning control (ILC), again with no estimation performance results. The identified model was applied with a PID-type adaptive controller and the control performance was guaranteed. Sun and He (2007) proposed a kind of two dimensional RLS for discrete time-varying system without any simulation or experimental result.

This paper focuses on online system identification, particularly recursive updating algorithms, overcomes the large parameters varying problem. It deals with the problem by imposing a penalty term in cost function to avoid such kind of parameter variation. Besides, a recursive identification is proposed and a necessary condition for convergence is given. Section 1 provides some background information about this paper. Section 2 gives the problem setup and derives the recursive identification. Section 3 gives some guidelines for the application in practical senario. In Section 4, a simulation result is provided. Section 5 discusses the result and draws conclusions.

2. PROBLEM FORMULATION

The main purpose of this paper is to solve the problem of parameters estimation varying for the two dimensional system identification. The basic idea is to impose a soft constraint to prevent that kind of things from happening and smooth the estimated parameters.

2.1 Problem Setup

Although majority of batch processes possess high nonlinearity, a lot of them can be approximated by a set of linear models with finite dimensions. Injection velocity vs. hydraulic valve opening is a vivid example to illustrate that. According to Yang (2004) research, it can be well approximated by a set of second-order ARX model along time direction. So here we narrow our scope on how to identify a time-varying ARX model as follows.

$$\begin{aligned} &y_k(t) + a_{1,0}(t)y_k(t-1) + a_{2,0}(t)y_k(t-2) + \cdots + a_{na,0}(t) \\ &y_k(t-na) = b_{1,0}(t)u_k(t-d) + b_{2,0}(t)u_k(t-d-1) + \ldots \\ &+ b_{nb,0}(t)u_k(t-d-nb) + w(t,k) \end{aligned} \tag{1}$$

where $y_k(t)$ and $u_k(t)$ are the system output and control input at time t and batch k respectively. And $a_{i,0}(t)$ and $b_{i,0}(t)$ are the system parameter on output and input at time t respectively, the subscript 0 denoting that it is the true parameter of the system. d represents the system delay, and $w(t,k)$ is zero-mean white noise with variance Q. Besides, na and nb are respectively standing for the order of output part and input part.

(1) can also be represented as

$$y_k(t) = \phi_k^T(t)\theta_0(t) + w(k,t) \tag{2}$$

where

$$\begin{aligned} \phi_k(t) = [&y_k(t-1) \quad y_k(t-2) \quad \ldots \quad y_k(t-na) \\ &u_k(t-d) \;\; u_k(t-d-1) \;\ldots\; u_k(t-d-nb)]^T \end{aligned} \tag{3}$$

and

$$\begin{aligned} \theta_0(t) = [&-a_{1,0}(t) \;\; -a_{2,0}(t) \;\ldots\; -a_{na,0}(t) \;\; b_{1,0}(t) \;\; b_{2,0}(t) \\ &\ldots \quad b_{nb,0}(t)]^T \end{aligned} \tag{4}$$

where T denotes the transpose.

Supposing that the system is running at time t and batch k, the best prediction of output $\hat{y}_k(t)$ is just the expectation of $y_k(t)$, which is

$$\begin{aligned} \hat{y}_k(t) &= E[y_k(t)|D] \\ &= E[\phi_k^T(t)\theta_0(t) + w(k,t)|D] \\ &= E[\phi_k^T(t)\theta_0(t)|D] && (w(k,t) \text{ is zero mean.}) \\ &= \phi_k^T(t)E[\theta_0(t)|D] && (\phi_k(t) \text{ is deterministic.}) \\ &= \phi_k^T(t)\hat{\theta}_k(t) \end{aligned} \tag{5}$$

where $\hat{\theta}_k(t)$ is the identified parameters of time t and batch k. D represents the past input and output data.

Notice that the system parameter vector $\theta_0(t)$ is only a function of time t, which means that if only look at a certain time spot, say it t, and from different batches, the system is actually a linear batch invariant system similar to the famous linear time invariant (LTI) system. In this way, define a cost function as follows.

$$\begin{aligned} J(t,k) &= \sum_{i=1}^{k} \|y_i(t) - \hat{y}_i(t)\|_2^2 \\ &= \sum_{i=1}^{k} \|y_i(t) - \phi_i^T(t)\hat{\theta}_k(t)\|_2^2 \end{aligned} \tag{6}$$

In reality, properties of lots of chemical engineering processes, which is associated with model, are slowly changing with time not undergoing a large scale change within a short period. So if the identified parameters vary intensively, it does not satisfy reality in most cases. If the cost function proposed above is simply applied, it may cause such a kind of problem because it takes a blind eye to the vary of identified parameters along time direction. On the other hand, the vary of estimated parameters will also cause large vary on the controller output, which influences on the output of the system in return. Based on the reasons above, it is necessary to impose a penalty term into (6). And it turns out to be

$$J(t,k) = \sum_{i=1}^{k} \|y_i(t) - \phi_i^T(t)\hat{\theta}_k(t)\|_2^2 + \|\hat{\theta}_k(t) - \hat{\theta}_k(t-1)\|_{A(t)}^2 \tag{7}$$

where $\|x\|_A = x^T A x$ and x is a column vector and A is a positive definite matrix.

Now, the system identification problem can be summarised as an optimisation problem like

$$\min_{\hat{\theta}_k(t)} J(t,k) \tag{8}$$

2.2 Recursive Form Derivation

In this part, we try to solve the optimisation problem above and turn the result into a recursive way so that online application can be implemented.

Firstly, take derivatives of $J(t,k)$ with respect to $\hat{\theta}_k(t)$, let it be equal to zeros, and then we have

$$\begin{aligned} \frac{\partial J(t,k)}{\partial \hat{\theta}_k(t)} = &-2\sum_{i=1}^{k} \phi_i(t)[y_i(t) - \phi_i^T(t)\hat{\theta}_k(t)] + 2A(t)[\hat{\theta}_k(t) \\ &- \hat{\theta}_k(t-1)] = 0 \end{aligned} \tag{9}$$

Solve (9), and yield that

$$\hat{\theta}_k(t) = [A(t)+\sum_{i=1}^{k}\phi_i(t)\phi_i^T(t)]^{-1}[A(t)\hat{\theta}_k(t-1)+\sum_{i=1}^{k}\phi_i(t)y_i(t)] \tag{10}$$

Observing (10), note that taking an inverse of a matrix has the computational complexity of $\mathcal{O}(n^3)$, if say Gauss-Jordan elimination approach. Therefore, a less computational and convenient way should be figured out.

Define

$$P_k^{-1}(t) = A(t) + \sum_{i=1}^{k}\phi_i(t)\phi_i^T(t) \tag{11}$$

and then we easily have

$$P_k^{-1}(t) = P_{k-1}^{-1}(t) + \phi_k(t)\phi_k^T(t) \tag{12}$$

Lemma 1. (Matrix inversion lemma). [Zhu (2001)] Let A, B, C and D be matrices with compatible dimensions and the inverses of A and C exist. Then

$$(A + BCD)^{-1} = A^{-1} - A^{-1}B(C^{-1} + DA^{-1}B)^{-1}DA^{-1} \tag{13}$$

Apply matrix inversion lemma to (12), and obtain that

$$\begin{aligned}
P_k(t) =& P_{k-1}(t) - P_{k-1}(t)\phi_k(t)[1 + \phi_k^T P_k(t)\phi_k(t)]^{-1}\phi_k^T(t) \\
& P_{k-1}(t) \\
=& P_{k-1}(t) - \frac{P_{k-1}(t)\phi_k(t)\phi_k^T(t)P_{k-1}(t)}{1 + \phi_k^T(t)P_{k-1}(t)\phi_k(t)} \\
& (\text{Now } 1 + \phi_k^T(t)P_{k-1}(t)\phi_k(t) \text{ becomes a scalar.})
\end{aligned} \tag{14}$$

For (10), get that

$$\begin{aligned}
\hat{\theta}_k(t) =& P_k(t)[A(t)\hat{\theta}_k(t-1) - A(t)\hat{\theta}_{k-1}(t-1) + \phi_k(t)y_k(t) \\
& + A(t)\hat{\theta}_{k-1}(t-1) + \sum_{i=1}^{k-1}\phi_i(t)y_i(t)] \\
=& P_k(t)[A(t)\hat{\theta}_k(t-1) - A(t)\hat{\theta}_{k-1}(t-1) + \phi_k(t)y_k(t) \\
& + P_{k-1}^{-1}(t)\hat{\theta}_{k-1}(t)] \\
=& P_k(t)\{[P_k^{-1}(t) - \phi_k(t)\phi_k(t)^T(t)]\hat{\theta}_{k-1}(t) + A(t) \\
& \hat{\theta}_k(t-1) - A(t)\hat{\theta}_{k-1}(t-1) + \phi_k(t)y_k(t)\}
\end{aligned} \tag{15}$$

The parameter identification update equation is

$$\begin{aligned}
\hat{\theta}_k(t) =& \hat{\theta}_{k-1}(t) + K_1[y_k(t) - \phi_k^T(t)\hat{\theta}_{k-1}(t)] + K_2[\hat{\theta}_k(t-1) \\
& - \hat{\theta}_{k-1}(t-1)]
\end{aligned} \tag{16}$$

where

$$K_1 = P_k(t)\phi_k(t) \tag{17}$$
$$K_2 = P_k(t)A(t) \tag{18}$$

Combining (14), (16), (17) and (18), we have the whole algorithm.

Remark: Observing (16), with note that if let $A(t)$ equal to zero, it turns out to be a traditional recursive least squares rotated vertically. The first and second terms on RHS can be interpreted as updating parameters from batch direction according to the novel perdition error. And the third term on RHS is feedback information from difference of identified parameters from last sampling time.

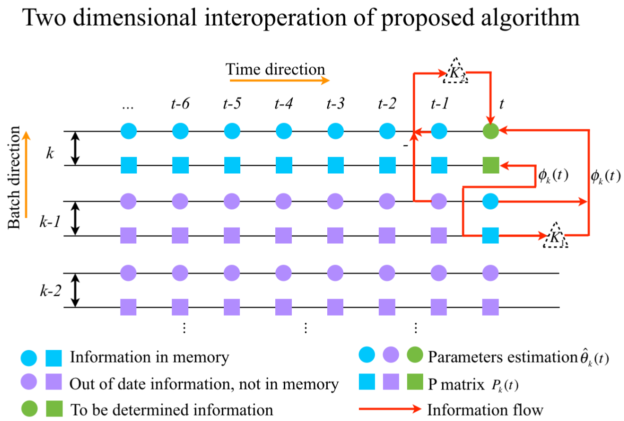

Fig. 1. Two dimensional information updating flow of proposed algorithm

It is essentially a kind of information from time direction. From (14), we know that it is solely updated along batch direction. All the information updating flow is illustrated as Fig. 1 shows.

3. ALGORITHM ANALYSIS

A necessary condition of convergence is presented in this section. Furthermore, a guideline of algorithm application is also provided.

3.1 Necessary Condition of Convergence

First, denote

$$\tilde{\theta}_k(t) = \theta_0(t) - \hat{\theta}_k(t) \tag{19}$$

Then (16) becomes

$$\begin{aligned}
\tilde{\theta}_k(t) =& \tilde{\theta}_{k-1}(t) - P_k(t)\phi_k(t)[y_k(t) - \phi_k^T(t)\hat{\theta}_{k-1}(t)] \\
& - P_k(t)A(t)[\hat{\theta}_k(t-1) - \hat{\theta}_{k-1}(t-1)] \\
=& \tilde{\theta}_{k-1}(t) - P_k(t)\phi_k(t)[\phi_k^T(t)\theta_0(t) + w(t,k) \\
& - \phi_k^T(t)\hat{\theta}_{k-1}(t)] - P_k(t)A(t)[\hat{\theta}_k(t-1) \\
& - \hat{\theta}_{k-1}(t-1)]
\end{aligned} \tag{20}$$

Take expectation on both sides and get

$$E[\tilde{\theta}_k(t)] = [1-P_k(t)\phi_k(t)\phi_k^T(t)]\tilde{\theta}_{k-1}(t)+P_k(t)A(t)\delta\tilde{\theta}_k(t-1) \tag{21}$$

where δ is batch-wise backward difference operator.

It is easy to see the sufficient condition for convergence is that

$$\rho(1 - P_k(t)\phi_k(t)\phi_k^T(t)) < 1 \tag{22}$$

and $\sum_{k=1}^{\infty} P_k(t)A(t)\delta\tilde{\theta}_k(t-1)$ is bounded. Where $\rho(X)$ stands for the largest absolute value of eigenvalue of matrix X.

3.2 Guideline for Algorithm Application

In the above derivation, $A(t)$ is a constant across batches, and it means that $A(t)$ should be fixed before the start of the whole identification process and not allowed to change. It is not favourable in practice. Note that only K_2 is related to $A(t)$ and the recursive form of $P_k(t)$ is the same to

RLS's except in different domains. Actually, $A(t)$ can be a dynamic matrix across batches and get tuned according to the situations of different batches. In spite of the mismatch between constant and dynamic weighting matrix, the influence of assuming constant $A(t)$ is negligible due to two reasons. One is $A_k(t)$ in most cases does not have a large change between two consecutive batches and is achievable. The other is that the main dynamics of $P_k^{-1}(t)$ and $\hat{\theta}_k(t)$ are dominant by the increment of $\phi_i(t)\phi_i^T(t)$ and prediction error. Thus, for the sake of flexibility, $A(t)$ can be replaced with $A_k(t)$ in application.

From the aspect of easy implementation, although $A_(t)$ is a positive definite matrix, it can even be selected as a diagonal matrix for simplicity. For this case, according to authors' experience, every entry along diagonal is selected from the range between 0 and 1, otherwise it may degrade identification performance. It is also noticed from simulation that input parameter estimations are much more sensitive to noise than that of output. Thus, larger weights on input parameters are recommended.

Another thing worth mentioning is how to give initial values for the second batch. Note that (11) is unrelated to the weighting matrix $A_(t)$, and actually $A_(t)$ just provides a initial value for $P_k(t)$. Of course, it is also allowed to pick up other suitable initial values for $P_k(t)$, but inherited from the first batch with conventional RLS is prohibited, owing to the fact that usually the initial value of $P_k(t)$ in the first several time spots is large and it will lead to a large updating gain leading to a bad transient process. It also reminds us to pick up a relatively small initial value for $P_k(t)$ to prevent a large updating gain.

4. SIMULATION

Injective moulding, a typical batch process, is a important polymer process technique with complicated mechanism involving a set of process variables to determine the final quality of products. Among all the process variables, injection velocity is a pivotal one, and self tuning regulator (STR) is proved to be a good candidate to control injection velocity, see Yang (2004). System identification is a core part of STR. Thus, a simulation based on injection velocity vs. hydraulic valve opening is presented in this section.

Consider the model as follows (Yang (2004)).

$$G(z) = \frac{1.69z^{-1} + 1.419z^{-2}}{1 - 1.582z^{-1} + 0.5916z^{-2}} \tag{23}$$

A time-varying model modified is

$$G(z) = \begin{cases} \dfrac{1.69z^{-1} + 1.419z^{-2}}{1 - 1.582z^{-1} + 0.5916z^{-2}} \\ (t \in [0, 150) \cup (300, 400]) \\ \dfrac{(1.69 - 0.2 * \frac{t-150}{150}) * z^{-1} + 1.419z^{-2}}{1 + (-1.58 + 0.2 * \frac{t-150}{150})z^{-1} + 0.5916z^{-2}} \\ (t \in [150, 300]) \end{cases} \tag{24}$$

The control input is designed as Pseudo-Random-Binary-Sequence (PRBS) to assure enough innovations. And the noise is taken as zero mean with variance 0.01 white noise. The signal-to-noise (SNR) of the system is 20dB.

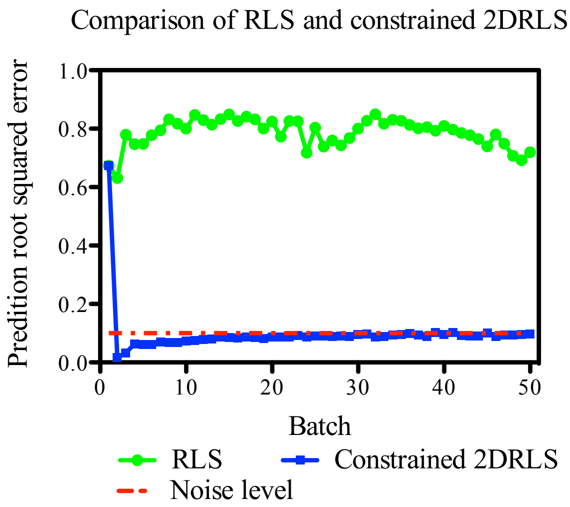

Fig. 2. Comparison of RLS and constrained 2DRLS on prediction error

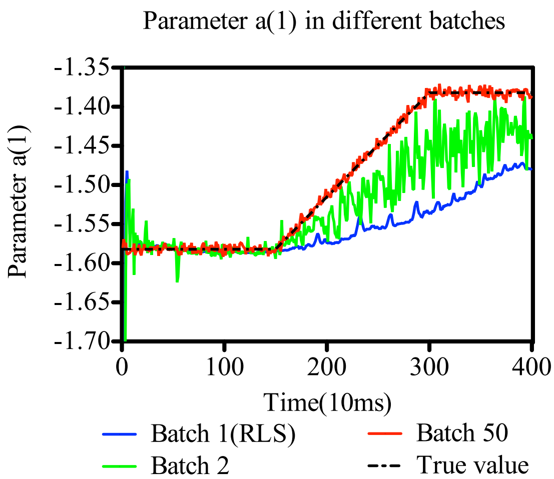

Fig. 3. Parameter a(1) in different batches

The system identification algorithm applied on the first batch is conventional RLS with a forgetting factor of 0.99. And the weight matrix $A(t)$ selected for simulation is

$$A(t) = \begin{bmatrix} 0.6 & & & \\ & 0.6 & & \\ & & 0.99 & \\ & & & 0.99 \end{bmatrix} \tag{25}$$

From Fig. (2), we can see that the performance of conventional RLS maintains a certain level as batches run, while the perdition error of constrained 2DRLS first decreases very fast which is a kind of over-fitting, and then gradually increases and reaches and maintains at that level which is the noise level and also should be the limit of all system identification algorithms based on ARX model.

$a(1)$ and $a(2)$ are two coefficients in $G(z)$'s numerator. And $b(1)$ and $b(2)$ are two coefficients in $G(z)$'s denominator. From Fig. (3) to Fig. (6), it is obvious that the identified parameters track the true parameters gradually as batches

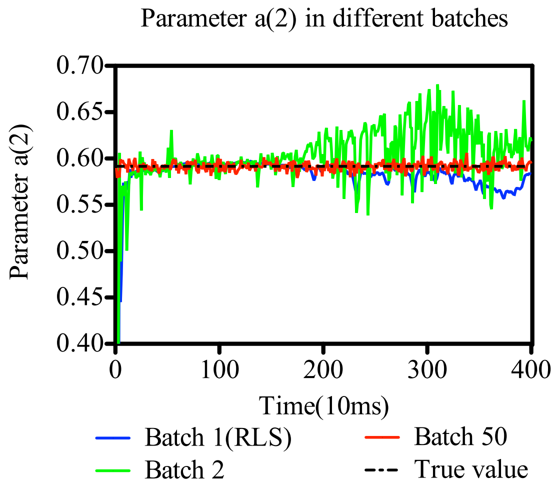

Fig. 4. Parameter a(2) in different batches

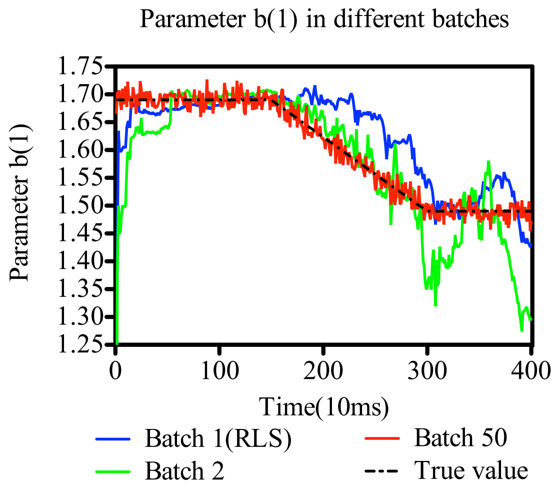

Fig. 5. Parameter b(1) in different batches

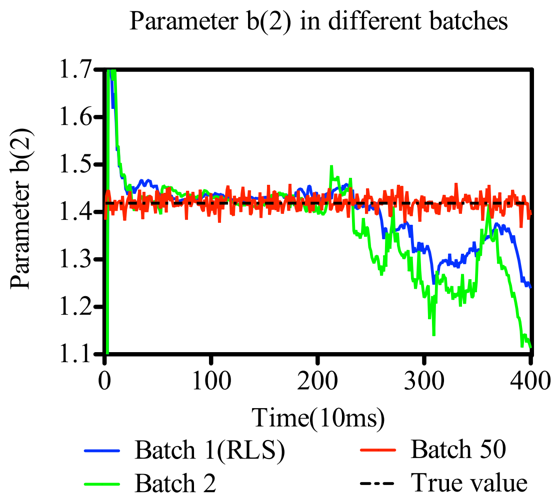

Fig. 6. Parameter b(2) in different batches

go on, al- though during the process, in some batches, the identified parameter suffers high-frequency variants.

5. CONCLUSION

This paper addresses a constrained two dimensional RLS system identification approach, which can track true parameters gradually as batches go on, meanwhile, also limit the identified parameters vary along time. A necessary condition of algorithm convergence is provided as well. The paper also sheds some light on how to apply the algorithm into practical scenario. A simulation on injection velocity model also illustrated the advantage of the proposed approach over the conventional ones.

ACKNOWLEDGEMENTS

This work is supported by the National Science Funds under Grant 61273145 and Hong Kong Research Grant Council under project No. 61252.

REFERENCES

Chi, R., Hou, Z., and Xu, J. (2008). Adaptive ILC for a class of discrete-time systems with iteration-varying trajectory and random initial condition. *Automatica*, 44(8), 2207–2213.

Eykhoff and Pieter (1974). *System identification : parameter and state estimation*. Wiley-Interscience, London; New York etc.

Ljung, L. (2009). *System identification: Theory for the user*. Prentice-Hall, Upper Saddle River, NJ.

Ma, D.L. and Braatz, R.D. (2003). Robust identification and control of batch processes. *Computers & Chemical Engineering*, 27(8–9), 1175–1184.

Mehra, R.K. and Lainiotis, D.G. (1976). *System identification : advances and case studies*. Academic Press, New York.

Shi, J., Gao, F., and 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*, 15(8), 907–924.

Shi, J., Gao, F., and Wu, T.J. (2007). Single-cycle and multi-cycle generalized 2D model predictive iterative learning control (2D-GPILC) schemes for batch processes. *Journal of Process Control*, 17(9), 715–727.

Söderström, T. and Stoica, P. (1989). *System identification*. Prentice Hall, New York.

Sun, M. and He, X. (2007). Iterative learning identification and control of discrete time-varying systems. In *Control Conference, 2007. CCC 2007. Chinese*, 520–524.

Tayebi, A. (2004). Adaptive iterative learning control for robot manipulators. *Automatica*, 40(7), 1195–1203.

Yang, Y. (2004). *Injection molding control : from process to quality*. Ph.D. thesis, Hong Kong University of Science and Technology.

Yang, Y. and Gao, F. (1998). Adaptive control of nozzle melt packing pressure. *Journal of intelligent material systems and structures*, 9(12), 1046–1050.

Yang, Y. and Gao, F. (1999). Cycle to cycle and within cycle adaptive control of nozzle pressure during packing-holding for thermoplastic injection molding. *Polymer Engineering & Science*, 39(10), 2042–2063.

Zhu, Y. (2001). *Multivariable system identification for process control*. Pergamon, Amsterdam.