Proceedings of the
44th IEEE Conference on Decision and Control, and
the European Control Conference 2005
Seville, Spain, December 12-15, 2005

Self-Tuning Iterative Learning Control for Time Variant Systems

Ali Madady

Abstract— **We consider the iterative learning control problem from an adaptive control viewpoint. The STILCS (self-tuning iterative learning control systems) problem is formulated in a general case, when the underlying repetitive linear process is time-variant and its parameters are all unknown, its initial conditions are not fixed and are not determinable in various iterations. A solution procedure is presented for this problem. The Lyapunov technique is employed to ensure the convergence of the presented STILCS. The computer simulation results are included to illustrate the effectiveness of the proposed STILCS.**

I. INTRODUCTION

ITERATIVE learning control (ILC) was originally proposed in the robotics community as an intelligent teaching mechanism for robot manipulators [1]. The principle of the iterative learning control systems (ILCS) may be stated as follows: at each execution of the control algorithm, some data as errors are recorded. These are used by the learning algorithm in the next execution for improving the control inputs and progressively reducing the output errors. After a number of repeated trials, the process should obtain an appropriate control input.

In the last two decades this field has attracted considerable research interest and achieves significant progress in both theory and application. The interested readers may refer to [2] and [3], two comprehensive books on ILCS survey.

In this paper, we consider the problem of designing controller for repetitive processes, where the parameters of processes are unknown. Similar to the traditional (un-repetitive) cases, in which the adaptive techniques are used to design the controller for unknown systems, we can use the adaptive control schemes to control the repetitive unknown systems. For this purpose some efforts are done to use the adaptive control approaches in iterative learning. In [4] an adaptive learning control, which consists of a linear open-loop control law and an adjustable scalar gain, was proposed to control a linear minimum-phase repetitive time-invariant continues single input - single output system. An adaptive robust ILC method based on dead-zone scheme was presented in [5] for control a class of continues time nonlinear uncertain systems. In [6] a non-linear iterative learning based on an adaptive Lyapunov technique was offered to control the discrete-time linear processes. A closed-loop linear adaptive learning algorithm was established in [7] for single input-single output time-invariant repetitive systems. In [8] a 2D system approach based on nonlinear adaptive control techniques was offered. The adaptive ILC for nonlinear systems with unknown high-frequency gain was analyzed in [9]. The LMI approach was studied for analysis and controller design for discrete linear repetitive processes in [10]. Also, the problem of parameter optimization in ILC was formulated and solved in [11]. In [12] a model reference adaptive ILC was presented for discrete-time linear systems. The problem of adaptive ILC for robot manipulators was solved in [13]. Three classes of learning algorithms, consists of the previous cycle learning (PCL), the current cycle learning (CCL) and the synergy previous and current cycle learning (PCCL), were analyzed and compared in [14]. In [15], an algebraic approach was considered to ILC.

However, none of the papers, which addresses linear time-varying unknown systems with variable initial conditions. In this paper, we deal with the learning control problem for a discrete linear time-varying unknown system, where the system parameters are all unknown and its initial conditions are variable in various iterations. That is the aim of this paper is to extend the self-tuning approach to repetitive cases when the underlying unknown processes are time-variant.

The paper is organized as follows. Section 2 formulates the STILCS (self-tuning iterative learning control systems) problem. In Section 3 we solve this problem. The convergence of the proposed self-tuning iterative learning procedure is analyzed in section 4. In Section 5 a numerical simulation example is given. Conclusion is deferred to Section 6.

II. PROBLEM FORMULATION

Suppose the underlying single input single output (SISO) discrete-time repetitive process described by:

$$\begin{aligned} \mathrm{x}(i+1,j) &= \mathrm{A}(i)\mathrm{x}(i,j)+\mathrm{b}(i)u(i,j) \quad && i=0,1,\ldots,M \\ y(i,j) &= \mathrm{c}^T(i-1)\mathrm{x}(i,j) && i=0,1,\ldots,M+1 \\ & && j=0,1,\ldots \end{aligned} \tag{1}$$

where $\mathrm{x}\in\Re^n$, $u\in\Re$ and $y\in\Re$ denote the state , the input and the output respectively. Integer independent variables i and j respectively denote the time variable and the operation or iterations number. Integer M is the time duration of iterations. $\mathrm{A}(i)$, $\mathrm{b}(i)$ and $\mathrm{c}(i-1)$ are real-valued coefficients with appropriate dimensions, and are also time-variant, but assumed to be the same from one trial to the next.

Author is with the Tafresh University (as a full time faculty member), Tafresh, IRAN (e-mail: drmadady@yahoo.com).

We define the STILCS problem as follows.

Consider (1) and make the following reasonable assumptions:

A1) All the process parameters, namely the matrix $\mathrm{A}(i)$ and the vectors $\mathrm{b}(i)$ and $\mathrm{c}(i)$, are unknown.

A2) The process initial conditions in each iteration ($\mathrm{x}(0,j)$) is not determinable and may be any random value.

A3) A collection of the desired output trajectories $y_d(i,j)$, which may be different for various iterations, is given.

A4) The state $\mathrm{x}(i,j)$ is accessible.

A5) The scalars $\mathrm{c}^T(i)\mathrm{b}(i)$ are all nonzero.

Determine the control input sequence $u(i,j)$, such that the following tracking can be established:

$$\lim_{j\to\infty}\left(y(i,j)-y_d(i,j)\right)=0 \quad \text{for } i=1,2,\ldots,M+1 \tag{2}$$

III. Soluition Procedure Of STILCS Problem

In the STILCS problem, we deal with controlling the repetitive processes with unknown parameters. Both adaptive control approaches, MRAC (model reference adaptive control) and STR (self-tuning regulators), are reported to have nice features for dealing with uncertainty. However, they are not well suited in handling repetitive trajectory tracking control tasks, since adaptive control does not utilize the knowledge that the operation is repetitive.

Here, we extend the self-tuning approach to repetitive trajectory tracking. For this purpose, the following closed-loop control law is proposed for determining $u(i,j)$ in (1):

$$\begin{gathered}u(i,j)=\mathrm{F}(i,j)\mathrm{x}(i,j)+G(i,j)y_d(i+1,j)\\ i=0,1,\ldots,M \qquad j=0,1,\ldots\end{gathered} \tag{3}$$

where $\mathrm{F}(i,j)\in\Re^{1\times n}$ and $G(i,j)\in\Re$ are adjustable gains, and we call them the *learning gains*.

Substituting for $u(i,j)$ from (3) into (1) yields:

$$\begin{aligned}\mathrm{x}(i+1,j)=&\left(\mathrm{A}(i)+\mathrm{b}(i)\mathrm{F}(i,j)\right)\mathrm{x}(i,j)\\ &+\mathrm{b}(i)G(i,j)y_d(i+1,j)\end{aligned} \tag{4}$$

Multiplying (4) from left side by $\mathrm{c}^T(i)$ implies:

$$\begin{aligned}y(i+1,j)=&\,\mathrm{c}^T(i)\left(\mathrm{A}(i)+\mathrm{b}(i)\mathrm{F}(i,j)\right)\mathrm{x}(i,j)\\ &+\mathrm{c}^T(i)\mathrm{b}(i)G(i,j)y_d(i+1,j)\end{aligned} \tag{5}$$

If we choose $\mathrm{F}(i,j)$ and $G(i,j)$ as follows:

$$\mathrm{F}(i,j)=\frac{-\mathrm{c}^T(i)\mathrm{A}(i)}{\mathrm{c}^T(i)\mathrm{b}(i)},\ G(i,j)=\frac{1}{\mathrm{c}^T(i)\mathrm{b}(i)} \tag{6}$$

Then we will have:

$$\begin{gathered}y(i,j)=y_d(i,j)\\ \text{for}\quad i=1,2,\ldots,M+1 \ \text{ and } \ j=0,1,\ldots\end{gathered} \tag{7}$$

But, $\mathrm{A}(i)$, $\mathrm{b}(i)$ and $\mathrm{c}(i)$ are unknown. Thus, first these should be estimated and then $\mathrm{F}(i,j)$ and $G(i,j)$ are determined according to the following equations:

$$\mathrm{F}(i,j)=\frac{-\hat{\mathrm{c}}^T(i,j)\hat{\mathrm{A}}(i,j)}{\hat{\mathrm{c}}^T(i,j)\hat{\mathrm{b}}(i,j)},\ G(i,j)=\frac{1}{\hat{\mathrm{c}}^T(i,j)\hat{\mathrm{b}}(i,j)} \tag{8}$$

where $\hat{\mathrm{A}}(i,j)$, $\hat{\mathrm{b}}(i,j)$ and $\hat{\mathrm{c}}(i,j)$ are respectively the estimations of $\mathrm{A}(i)$, $\mathrm{b}(i)$ and $\mathrm{c}(i)$ in the iteration j.

The next step is the establishing an online adaptive algorithm for estimating $\mathrm{A}(i)$, $\mathrm{b}(i)$ and $\mathrm{c}(i)$. For this purpose (1) is written as following compact form:

$$\Upsilon(i,j)=\theta(i)\mathrm{z}(i,j)\ \ i=0,1,\ldots,M\ \ j=0,1,\ldots \tag{9}$$

where:

$$\Upsilon(i,j)=\begin{bmatrix}\mathrm{x}(i+1,j)\\ y(i+1,j)\end{bmatrix},\mathrm{z}(i,j)=\begin{bmatrix}\mathrm{x}(i,j)\\ u(i,j)\\ \mathrm{x}(i+1,j)\end{bmatrix}$$
$$\theta(i)=\begin{bmatrix}\mathrm{A}(i) & \mathrm{b}(i) & 0\\ 0 & 0 & \mathrm{c}^T(i)\end{bmatrix} \tag{10}$$

The relation (9) has two-dimensional regressor form, which is introduced in [16]. According to the procedure of adaptive parameter estimation for two-dimensional systems, which is discussed in [16], a suitable algorithm is proposed as follows, for estimating the components of $\theta(i)$ in (9):

$$\hat{\mathrm{A}}(i,j+1)=\hat{\mathrm{A}}(i,j)-\mu(i,j)\mathrm{P}_1\begin{bmatrix}\mathrm{e_x}(i,j)\\ e_y(i,j)\end{bmatrix}\mathrm{x}^T(i,j) \tag{11a}$$

$$\hat{\mathrm{b}}(i,j+1)=\hat{\mathrm{b}}(i,j)-\mu(i,j)\mathrm{P}_1\begin{bmatrix}\mathrm{e_x}(i,j)\\ e_y(i,j)\end{bmatrix}u(i,j) \tag{11b}$$

$$\hat{\mathrm{c}}^T(i,j+1)=\hat{\mathrm{c}}^T(i,j)-\mu(i,j)\mathrm{P}_2\begin{bmatrix}\mathrm{e_x}(i,j)\\ e_y(i,j)\end{bmatrix}\mathrm{x}^T(i+1,j) \tag{11c}$$

$$i=0,1,\ldots,M \qquad j=0,1,\ldots$$

where:

$$\begin{aligned} &\mathrm{e_x}(i,j) = \hat{\mathrm{A}}(i,j)\mathrm{x}(i,j) + \hat{\mathrm{b}}(i,j)u(i,j) - \mathrm{x}(i+1,j) \\ &e_y(i,j) = \hat{\mathrm{c}}^T(i,j)\mathrm{x}(i+1,j) - y(i+1,j) \\ &\qquad i = 0,1,\ldots,M \qquad j = 0,1,\ldots \end{aligned} \tag{12}$$

Also, T denotes the transpose, $\mu(i,j)$ is a positive scalar namely the algorithm step size, P_1 and P_2 are respectively the n first rows, and the last row of an arbitrary symmetric positive definite matrix P, that is:

$$\mathrm{P} = \begin{bmatrix}\mathrm{P}_1\\ \mathrm{P}_2\end{bmatrix}, \mathrm{P}_1 \in \Re^{n\times(n+1)}, \mathrm{P}_2 \in \Re^{1\times(n+1)}, \mathrm{P} = \mathrm{P}^T > 0 \tag{13}$$

The control law (3), algebraic equations (8) and the adjusting algorithm (11) are the main parts of the presented STILCS.

IV. Convergence Analysis

The following definition is presented:

Definition- The proposed STILCS is said to be convergent, if for any initial conditions $\mathrm{x}(0,j)$, it generates an input sequence $u(i,j)$ for process (1), such that (2) holds.

Theorem - The presented STILCS is convergent if the step size $\mu(i,j)$ in the adjusting algorithm (11) is chosen in the following interval:

$$0 < \mu(i,j) < \frac{2}{\max\left(r_1(i,j), r_2(i,j)\right)\lambda_{\max}(\mathrm{P})} \tag{14}$$

where

$$\begin{aligned} r_1(i,j) &= \mathrm{x}^T(i,j)\mathrm{x}(i,j) + u^2(i,j) \\ r_2(i,j) &= \mathrm{x}^T(i+1,j)\mathrm{x}(i+1,j) \end{aligned}$$

and $\lambda_{\max}(\mathrm{P})$ denotes the largest eigenvalue of P.

proof:

We consider $V(i,j)$ as follows for candidate of Lyapunov function:

$$\begin{aligned} V(i,j) = Trace&\left[\tilde{\mathrm{A}}^T(i,j)\tilde{\mathrm{A}}(i,j)\right] \\ &+ \tilde{\mathrm{b}}^T(i,j)\tilde{\mathrm{b}}(i,j) + \tilde{\mathrm{c}}^T(i,j)\tilde{\mathrm{c}}(i,j) \\ &i = 0,1,\ldots,M \qquad j = 0,1,\ldots. \end{aligned} \tag{15}$$

where *Trace* is the sum of the diagonal elements of the matrix, and:

$$\tilde{\mathrm{A}}(i,j) = \hat{\mathrm{A}}(i,j) - \mathrm{A}(i), \tilde{\mathrm{b}}(i,j) = \hat{\mathrm{b}}(i,j) - \mathrm{b}(i)$$
$$\tilde{\mathrm{c}}(i,j) = \hat{\mathrm{c}}(i,j) - \mathrm{c}(i)$$

Change of $V(i,j)$ is defined as:

$$\begin{aligned} \Delta V(i,j) = V(i,j+1) - V(i,j) \\ i = 0,1,\ldots,M \ , \ j = 0,1,\ldots \end{aligned} \tag{16}$$

Evaluating the $\Delta V(i,j)$ one obtains:

$$\begin{aligned} \Delta V(i,j) = -\mathrm{e}^T(i,j)\mathrm{H}(i,j)\mathrm{e}(i,j) \\ i = 0,1,\ldots,M \quad , \quad j = 0,1,\ldots \end{aligned} \tag{17}$$

where: $\mathrm{e}(i,j) = \begin{bmatrix}\mathrm{e_x}(i,j)\\ e_y(i,j)\end{bmatrix}$

and $\mathrm{H}(i,j)$ is the following symmetric matrix:

$$\mathrm{H}(i,j) = 2\mu(i,j)\mathrm{P} - \mu^2(i,j)\mathrm{P}\mathrm{Q}(i,j)\mathrm{P} \tag{18}$$

$\mathrm{Q}(i,j) \in \Re^{(n+1)\times(n+1)}$ is as follows:

$$\mathrm{Q}(i,j) = \begin{bmatrix} r_1(i,j)\mathrm{I}_n & 0 \\ 0 & r_2(i,j)\end{bmatrix} \tag{19}$$

($\mathrm{I}_n \in \Re^{n\times n}$ is the identity matrix)

It is easy to show that if $\mu(i,j)$ is in the interval (14), then the symmetric matrix $\mathrm{H}(i,j)$ will be positive definite and we will have:

$$\Delta V(i,j) \le 0 \quad i = 0,1,\ldots,M \ , \ j = 0,1,\ldots \tag{20}$$

Thus:

$$\lim_{j\to\infty} \Delta V(i,j) = 0 \quad \text{for} \quad i = 0,1,\ldots,M \tag{21}$$

Or:

$$\lim_{j\to\infty} \mathrm{e_x}(i,j) = 0, \lim_{j\to\infty} e_y(i,j) = 0 \quad i = 0,1,\ldots,M \tag{22}$$

From (22) is concluded that for sufficiently large j, we have:

$$\mathrm{e_x}(i,j) = 0, e_y(i,j) = 0 \quad \text{for} \quad i = 0,1,\ldots,M \tag{23}$$

Therefore, from (11) the constant values relative to variable j are resulted for $\hat{A}(i,j)$, $\hat{b}(i,j)$ and $\hat{c}(i,j)$, like $\hat{A}_i$, $\hat{b}_i$ and $\hat{c}_i$ respectively, that is:

$$\hat{A}(i,j)=\hat{A}_i \;,\; \hat{b}(i,j)=\hat{b}_i \;,\; \hat{c}(i,j)=\hat{c}_i \quad \text{for } i=0,1,\dots,M \text{ and sufficiently large } j \tag{24}$$

From (12), (23) and (24), the following relations can be concluded:

$$x(i+1,j)=\hat{A}_i x(i,j)+\hat{b}_i u(i,j) \tag{25}$$

$$y(i+1,j)=\hat{c}_i^T x(i+1,j) \tag{26}$$

$$\text{for } i=0,1,\dots,M \text{ and sufficiently large } j$$

On the other hand (3), (8) and (24) imply:

$$u(i,j)=-\frac{\hat{c}_i^T\hat{A}_i}{\hat{c}_i^T\hat{b}_i}x(i,j)+\frac{1}{\hat{c}_i^T\hat{b}_i}y_d(i+1,j) \quad \text{for } i=0,1,\dots,M \text{ and sufficiently large } j \tag{27}$$

Substituting for $u(i,j)$ from (27) into (25) yields:

$$x(i+1,j)=\left(\hat{A}_i-\frac{\hat{b}_i\hat{c}_i^T\hat{A}_i}{\hat{c}_i^T\hat{b}_i}\right)x(i,j)+\frac{\hat{b}_i}{\hat{c}_i^T\hat{b}_i}y_d(i+1,j) \quad \text{for } i=0,1,\dots,M \text{ and sufficiently large } j \tag{28}$$

Multiplying (28) from left side by $\hat{c}_i^T$ implies:

$$\hat{c}_i^T x(i+1,j)=y_d(i+1,j) \quad \text{for } i=0,1,\dots,M \text{ and sufficiently large } j \tag{29}$$

From (26) and (29) we have:

$$y(i,j)=y_d(i,j) \quad \text{for } i=1,2,\dots,M+1 \text{ and sufficiently large } j \tag{30}$$

That means:

$$\lim_{j\to\infty} y(i,j)=y_d(i,j) \quad i=1,2,\dots,M+1 \tag{31}$$

Here the proof of Theorem is completed.

■

V. SIMULATION RESUTS

In order to illustrate the performance of the proposed STILCS procedure, a numerical example is presented.

Consider the following time-variant second order repetitive process:

$$\begin{bmatrix} x_1(i+1,j) \\ x_2(i+1,j) \end{bmatrix} = A(i)\begin{bmatrix} x_1(i,j) \\ x_2(i,j) \end{bmatrix} + b(i)u(i,j)$$

$$i=0,1,\dots,19$$

$$y(i,j)=c^T(i-1)\begin{bmatrix} x_1(i,j) \\ x_2(i,j) \end{bmatrix}$$

$$i=0,1,\dots,20 \quad j=0,1,\dots$$

where:

$$A(i)=\begin{bmatrix} 1+\frac{i}{19} & 2\times(-1)^i \\ 2^{-i} & 1.2\sin(\frac{\pi i}{20}) \end{bmatrix}$$

$$b(i)=\begin{bmatrix} 2+\cos(\frac{\pi i}{10}) \\ 0 \end{bmatrix}, \; c^T(i)=\begin{bmatrix} i+1 & \frac{i}{i+10} \end{bmatrix}$$

The matrix $A(i)$ and the vectors $b(i)$, $c(i)$ are assumed to be unknown, and also the process initial conditions are supposed to be varied during the various iterations as follows:

$$x_1(0,j)=1+\sin(\frac{\pi j}{50}) \;,\; x_2(0,j)=(-0.5)^j$$

The desired output trajectory, which is shown in Fig. 1, is as follows for all iterations:

$$y_d(i,j)=(i-10.5)^2 \quad 1\le i\le 20$$

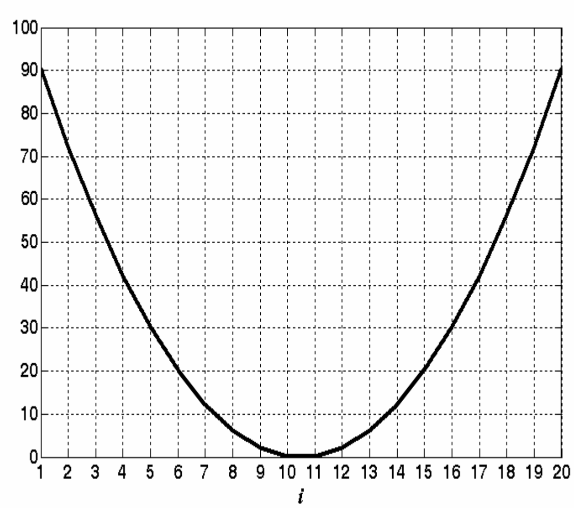

Fig. 1- The desired output trajectory

The matrix P is chosen as identity matrix in the adjusting algorithm (11), and considering (14), $\mu(i, j)$ is selected as follows:

$$\mu(i, j) = \frac{1}{\max\left(r_1(i, j), r_2(i, j)\right)\lambda_{\max}(\mathrm{P})}$$

The initial conditions of the algorithm (11), which is the prior estimations of the process parameters, are taken as follows:

$$\hat{\mathrm{A}}(i,0) = \begin{bmatrix} 1.5 & 0 \\ 0.5 & 1 \end{bmatrix}, \hat{\mathrm{b}}(i,0) = \begin{bmatrix} 2 \\ 1 \end{bmatrix}, \hat{\mathrm{c}}^{\mathrm{T}}(i,0) = \begin{bmatrix} 5 & 1 \end{bmatrix}$$

The obtained trajectories for process output are shown in Figs. 2- 6 in some iteration. These results demonstrate that with increasing the number of iterations, the process output progressively became close to the desired output trajectory.

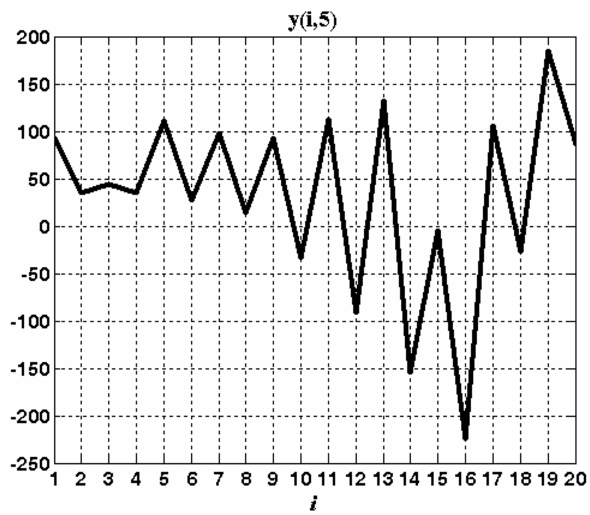

Fig. 2- The process output in iteration j=5

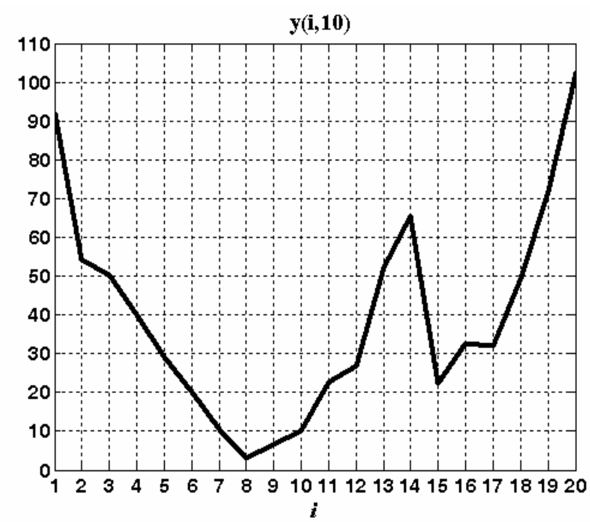

Fig. 3- The process output in iteration j=10

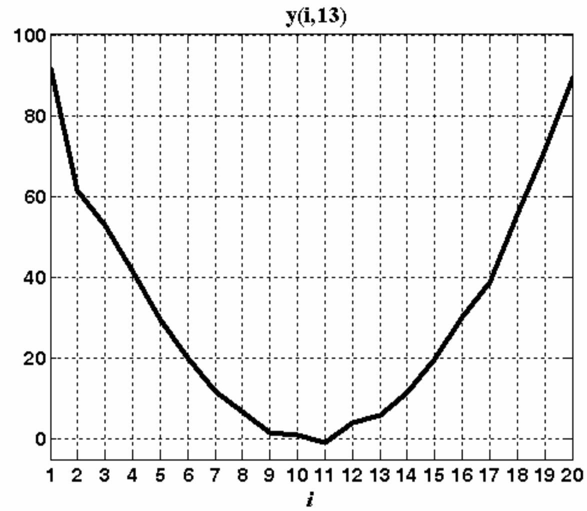

Fig. 4- The process output in iteration j=13

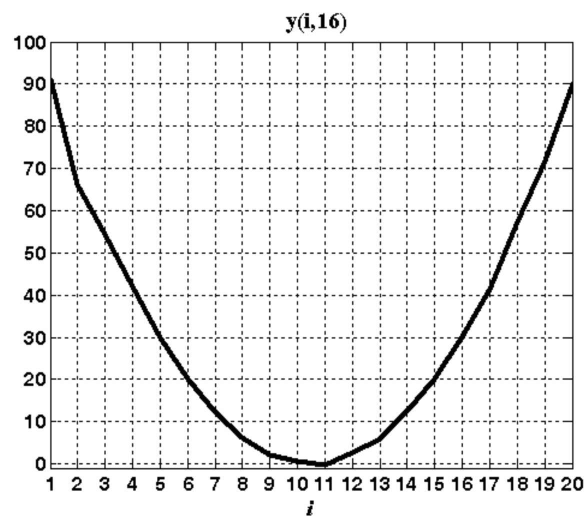

Fig. 5- The process output in iteration j=16

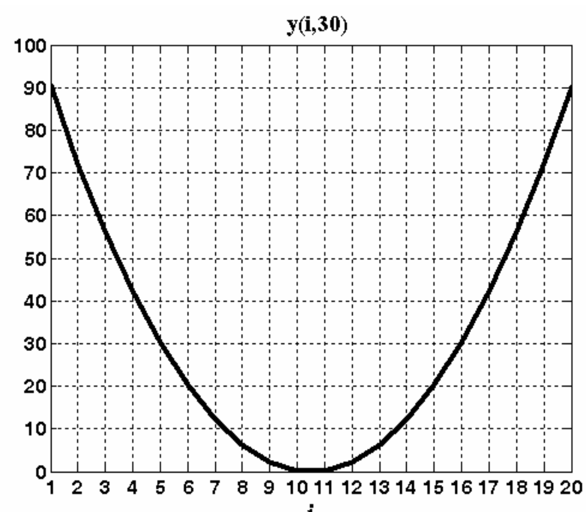

Fig. 6- The process output in iteration j=30

The *total learning error* in each iteration, which is defined below, is shown in Fig. 7.

$$E(j) = \sum_{i=1}^{M+1} (y(i,j) - y_d(i,j))^2 \qquad j = 0,1,...$$

It is obvious, that with increasing the number of the iterations, $E(j)$ is monotonically vanished.

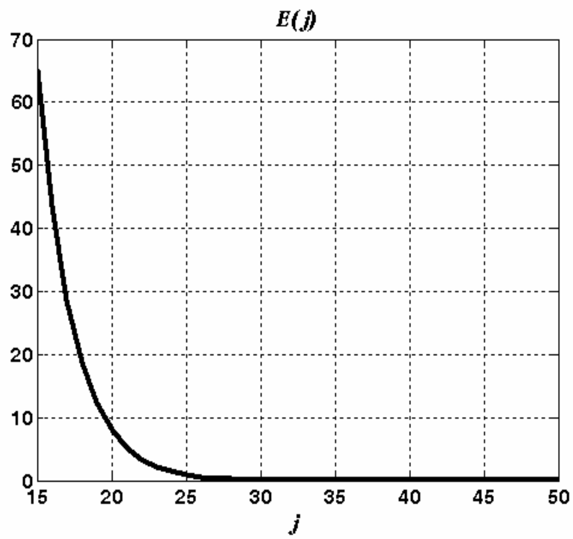

Fig. 7- The total learning error

VI. Conclusion

This paper has extended the self-tuning control approach to repetitive processes. The STILCS problem is formulated in a general case, when the underlying repetitive linear process is possibly time-variant and its parameters are all unknown and its initial conditions is randomly. A solution procedure is presented for this problem. In this solution procedure, the process input is assigned according to a closed-loop control law incorporating the process state and the desired output trajectory. There exist some feedback and feed forward adjustable gains in this control law, which are called the learning gains. An algorithm is proposed for adjusting the learning gains, so that the learning gains are modified in each repetition, utilizing the input-output data of the controlled process in previous operation. There exists an easily selectable factor, namely the algorithm step size, in this adjustment algorithm. The convergence of the presented STILCS procedure is analyzed by Lyapunov approach, and convergence condition is obtained in terms of algorithm step size range.

Although, the single-input and single-output case was studied, but the obtained results easily can be extended to the multi-input and multi-output case.

However, there are a number of problems, which merit further research, and these are subjects of our future work. Two such research problems are as follows:

1) The analysis of the convergence rate and the transient behavior of the above STILCS procedure.

2) Solving the STILCS problem when the state in the underlying process is not accessible but only the output of the process is measurable.

References

[1] S. Arimoto, S. Kawamura, and F. Miyazaki, “Bettering operation of robots by learning,” *Journal of Robotic Systems*, vol. 1(1), pp. 123–140, 1984.

[2] Z. Bien, and J. –X. Xu, *Iterative Learning Control: Analysis, Design, Integration and Applications*. Boston: Kluwer Academic Publishers, 1999.

[3] J. –X. Xu, and Y. Tan, *Linear and Nonlinear Iterative Learning Control*. New York: Springer-Verlag, 2003.

[4] D. H. Owens, and G. Munde, “Adaptive iterative learning control,” in *Proceedings of the 1996 IEE Colloquium on Adaptive Control*, London, UK, Jun 1996.

[5] J. –X. Xu and B. Viswanathan, “Adaptive robust iterative learning control with dead zone scheme,” *Automatica,* vol. 36, pp.91–99, 2000.

[6] M. French, and E. Rogers, “Non-linear iterative learning by an adaptive Lyapunov technique,” *International Journal of Control*, vol. 73, no. 10, pp. 840–850, 2000.

[7] D. H. Owens, and G. Munde, “Error convergence in an adaptive iterative learning controller,” *International Journal of Control*, vol. 73, no. 10, pp. 851–857, 2000.

[8] M. French, E. Rogers, H. Wibowo, and D.H. Owens, “A 2D systems approach to iterative learning control based on nonlinear adaptive control techniques,” in *Proceedings of the IEEE International Symposium on Circuits and Systems*, 2001, pp. II429–II432.

[9] H. – D. Chen, and P. Jiang, “Adaptive iterative learning control for nonlinear systems with unknown high-frequency gain,” *Kongzhi yu Juece/Control and Decision*, vol. 17, pp. 715–718, November 2002.

[10] K. Galkowski, E. Rogers, S. Xu, J. Lam, and D. H. Owens, “LMIs—A fundamental tool in analysis and controller design for discrete linear repetitive processes,” *IEEE Trans. On Circ. and Sys.—I: Fundamental Theory and Applications*, vol. 49, no. 6, pp. 768–778, June 2002.

[11] D. H. Owens, and K. Feng, “Parameter optimization in iterative learning control,” *International Journal of Control*, vol. 76, no. 11, pp. 1059–1069, 2003.

[12] A. Madady, and M. Shafiee, “Adaptive iterative learning control systems”, in *Proceedings of the Iranian Conference on Electrical Engineering* (ICEE2003), Shiraz, IRAN, May 2003, vol. 3, pp. 115–123.

[13] A. Tayebi, “Adaptive iterative learning control for robot manipulators,” *Automatica*, vol. 40, pp.1195–1203, 2004

[14] J. –X. Xu, T. H. Lee, and H. –W. Zhang, “Analysis and comparison of iterative learning control schemes,” *Engineering Applications of Artificial Intelligence*, vol. 17, pp.675–686, 2004

[15] J. J. Hatonen, D. H. Owens, and K. L. Moore, “An algebraic approach to iterative learning control,” *International Journal of Control*, vol. 77, no. 1, pp. 45–54, 2004.

[16] A. Madady, and M. Shafiee, “Two-dimensional adaptive parameter estimation,” in *Proceedings of the European Control Conference (ECC 2003),* University of Cambridge, Cambridge, UK , September 2003.