Proceedings of the American Control Conference
Anchorage, AK May 8-10, 2002

PI-type Iterative Learning Control Revisited [1]

YangQuan Chen and Kevin L. Moore
Center for Self-Organizing and Intelligent Systems (CSOIS)
Dept. of Electrical and Computer Engineering
UMC 4160, College of Engineering, 4160 Old Main Hill
Utah State University, Logan, UT 84322-4160, USA

Abstract

Iterative learning control (ILC) is a *value-added block* to enhance the feedback control performance by utilizing the fact that the system is operated repeatedly for the same task. In terms of how to use the tracking error signal of previous iteration to form the control signal of current iteration, ILC updating schemes can be classified as P-type, D-type, PI-type and PID type etc. So far, no one answered this question: *what's the use of the error integral in ILC updating law?* In this paper, for discrete-time linear time invariant systems, we show that the error integral in ILC updating scheme is helpful in achieving a monotonic convergence in a suitable norm topology other than the exponentially weighted sup-norm. Optimal design of PI-type ILC scheme is presented. Two extreme cases are considered to show that PI-type ILC can be better than P-type ILC in terms of convergence speed. Simulation results are presented to illustrate how to optimal design the PI-type ILC. We also show that when the number of time instants in an iteration is large, I-component in ILC updating law is of little use.
Keywords: Iterative learning control; monotonic convergence; proportional plus integral (PI) type ILC, optimal design.

1 Introduction

Iterative learning control (ILC) [1, 2] is proposed to enhance the feedback control performance by utilizing the fact that the system is operated repeatedly for the same task. While the formal mathematically rigorous analysis is initially due to [1], the basic idea can be traced back to [3] and even to [4] which is commented in [5]. Detailed literature reviews and recent developments on ILC research can be found in [6, 7, 8].

The original ILC scheme proposed in [1] is a D-type, i.e.,

$$u_{k+1}(t) = u_k(t) + \gamma \frac{\mathrm{d}}{\mathrm{d}t} e_k(t) \qquad (1)$$

where k is the iteration number and time $t \in [0, t_f]$; $u_k(t)$ is the control signal at the k-th iteration; $e_k(t)$ is the tracking error at the k-th iteration which is defined as $e_k(t) = y_d(t) - y_k(t)$ with $y_d(t)$ the desired output trajectory and $y_k(t)$ the system output at the k-th iteration; γ is the learning gain to be designed. This scheme works well for systems with relative degree one in the sense that $y_k(t) \to y_d(t)$ as $k \to \infty$, if $y_k(0) = y_d(0)$, $\forall t \in [0, t_f]$, provided that

$$|1 - \gamma h_1| < 1 \qquad (2)$$

where h_1 is the first Markov parameter. In the state space setting (A, B, C), $h_1 = CB \neq 0$. For systems with relative degree r, $h_r = CA^{r-1}B$ and the learning updating law (1) should be

$$u_{k+1}(t) = u_k(t) + \gamma \frac{\mathrm{d}^r}{\mathrm{d}t^r} e_k(t). \qquad (3)$$

For simplicity of our presentation, in the sequel, we use $r = 1$. A somewhat amazing fact is that, e.g. for $r = 1$, the ILC works without bothering to know the system matrix A. That is why ILC is attractive in application since we need less information about the system to be controlled. In general, a PID-type ILC scheme can be written as

$$\begin{aligned} u_{k+1}(t) &= u_k(t) + k_P e_k(t) + k_I \int_0^t e_k(\tau)\mathrm{d}\tau \\ &\quad + \gamma \frac{\mathrm{d}}{\mathrm{d}t} e_k(t) \end{aligned} \qquad (4)$$

where k_P, k_I and γ are PID learning gains. According to the existing literature, the ILC convergence condition is *still* the same as given in (2). This is another amazing fact.

The role of P-component (k_P) in a continuous-time D-type ILC scheme was studied in [9] where it is shown that a PD-type ILC scheme with properly chosen learning gains can be robust to non-zero initialization errors. However, the role of I-component (k_I) has not been studied in the literature to our best knowledge. In this paper, a PI-type ILC scheme is studied in discrete-time domain. This is due to the fact that ILC is essentially a memory based scheme (we need to store the signals of the previous iteration) and it is more natural to discuss ILC in a discrete-time domain. Our focus here is to investigate how the integral of tracking error affects the monotonic convergence of the iterative learning process.

[1]This work is supported in part by U.S. Army Automotive and Armaments Command (TACOM) Intelligent Mobility Program (agreement no. DAAE07-95-3-0023. Corresponding author: Dr YangQuan Chen. E-mail: yqchen@ieee.org; Tel. 01-435-7970148; Fax: 01-435-7972003. URL: http://www.crosswinds.net/~yqchen.

It is observed in [10] that although the λ-norm of tracking error from iteration to iteration can be proved to decay monotonically, the ∞-norm or sup-norm may increase to a huge value before it converges to the desired level. This transient behavior, which is a serious concern in the practical application of ILC schemes, can be improved by using an exponentially decay learning gain as discussed in [10]. One may argue that to make the convergence monotonic in sup-norm or 2-norm, one can use a high-gain feedback [11, 12]. However, this is not practical because the high-gain feedback may saturate the actuators. The fact that in some ILC schemes the error can grow quite large before converging has also been qualitatively discussed in [13, 14] from a frequency domain perspective. The effect can be explained as a result of the propagation of high-frequency components of the error by the ILC algorithm. Recently, in time domain, a condition for monotonic convergence of ∞-norm of tracking errors is established in [15]. There are some analysis results for monotonic convergence of ILC schemes via using approximate impulse response [16], reduced sampling rate [17] and for sampled data nonlinear systems [18]. How to achieve the monotonic convergence for discrete time systems via a proper ILC updating law design is addressed in a recent work [19] where a time varying learning gain is used to achieve monotonic learning convergence. In this paper, we will show that a PI-type ILC scheme will be more helpful in achieving the monotonic convergence of learning process than P-type alone. However, even under the optimal design (Sec. 3) of PI-type ILC scheme, as verified in the two extreme cases (Sec. 4), it is shown that when the number of time instants in an iteration N (or, t_f in continuous-time case; for sampled data systems, $t_f = NT_{\text{sample}}$) is large, I-component in ILC updating law is of *little* use, with some surprise.

The rest of this paper is organized as follows. In Sec. 2, some notations and preliminaries are given. In Sec. 3, monotonic convergence condition of PI-type ILC scheme is analyzed together with an optimal design method. In Sec. 4, two extreme cases are presented to show that the optimal PI-type ILC scheme converges faster than the optimal P-type scheme. Sec. 5 presents several simulation examples to demonstrate the helpful role of the integral of tracking error in achieving ILC monotonic convergence when N is smaller. When N is large, I-component is shown to be practically of no help. Finally, conclusions are presented in Sec. 6.

2 Notations and Preliminaries

Let an operation, or trial, of the system to be controlled be denoted by subscript "k" and let time during a given trial be denoted by "t," where $t \in [0, N]$. Each time the system operates the input to the system, $u_k(t)$, is stored, along with the resulting system error, $e_k(t) = y_d(t) - y_k(t)$, where $y_d(t)$ is the desired output. The plant to be controlled is a discrete-time, linear, time-invariant system of the form using $\mathcal{Z}$-transfer function:

$$\begin{aligned} Y(z) &= H(z)U(z) \\ &= (h_d z^{-1} + h_{d+1} z^{-2} + \cdots)U(z), \end{aligned} \tag{5}$$

where z^{-1} is the standard delay operator in time, and the parameters h_i are the standard Markov parameters of the system $H(z)$. We will also assume the standard ILC reset condition: $y_k(0) = y_d(0) = y_0$ for all k. If we define the "supervectors" [20]

$$U_k = [u_k(0), u_k(1), \cdots, u_k(N-1)]^T,$$

$$Y_k = [y_k(1), y_k(2), \cdots, y_k(N)]^T,$$

$$Y_d = [y_d(1), y_d(2), \cdots, y_d(N)]^T$$

and

$$E_k = [e_k(1), e_k(2), \cdots, e_k(N)]^T,$$

then the system can be written as

$$Y_k = H_p U_k, \tag{6}$$

where H_p is the matrix of Markov parameters of the plant, given by

$$H_p = \begin{bmatrix} h_1 & 0 & 0 & \dots & 0 \\ h_2 & h_1 & 0 & \dots & 0 \\ h_3 & h_2 & h_1 & \dots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ h_N & h_{N-1} & h_{N-2} & \dots & h_1 \end{bmatrix}. \tag{7}$$

For this system, the learning controller's goal is to derive an optimal input $u^*(t)$, for $t \in [0, N-1]$ by evaluating the error $e_k(t) = y_d(t) - y_k(t)$ on the interval $t \in [1, N]$. This is accomplished by adjusting the input from the current trial (u_k) to a new input (u_{k+1}) for the next trial. This adjustment is done according to an appropriate algorithm. The so-called Arimoto-type discrete-time ILC algorithm is given by

$$u_{k+1}(t) = u_k(t) + \gamma e_k(t+1) \tag{8}$$

where γ is the constant learning gain. We call this scheme P-type since the derivative information of the tracking error is not explicitly used. However, (8) should be understood as the *discrete-time counterpart* of the continuous-time D-type ILC (1).

The convergence properties of the Arimoto-type ILC algorithm have been well-established in the literature. Using a contraction mapping approach it is easy to see that the ILC scheme converges if the induced operator norm satisfies

$$\|I - \gamma H_p\|_i < 1. \tag{9}$$

Note that this sufficient condition ensures monotone convergence in the sense of the relevant norm topology. It is also possible to give the following necessary and sufficient condition for convergence [20]:

$$|1 - \gamma h_1| < 1. \tag{10}$$

Unfortunately, this second condition, similar to the condition given in (2) does *not* guarantee monotone convergence as observed in [15]. In addition to the necessary and sufficient condition for convergence (10), the other conditions to guarantee the monotone convergence can be found in a theorem given in [15], i.e.,

$$|h_1| > \sum_{j=2}^{N} |h_j|. \qquad (11)$$

This is just a sufficient condition which may be too restrictive since it does not relate to the learning gain γ.

In what follows, we will focus on the induced norm condition (9) using optimization techniques based on the available tuning knobs, e.g., γ in P-type scheme (8). In a PI-type scheme, an extra tuning knob is introduced. One may intuitively expect a better situation in achieving monotonic convergence using PI-type scheme over P-type scheme. We will confirm this expectation via a rigorous analysis.

3 PI-type ILC and Its Optimal Design

The PI-type ILC is given by

$$u_{k+1}(t) = u_k(t) + \gamma e_k(t+1) + k_I \sum_{j=1}^{t+1} e_k(j) \qquad (12)$$

where γ and k_I are PI learning gains. Using supervector representation, we can write

$$U_{k+1} = U_k(t) + \gamma E_k + k_I T_I E_k \qquad (13)$$

where

$$T_I = \begin{bmatrix} 1 & 0 & 0 & \dots & 0 & 0 \\ 1 & 1 & 0 & \dots & 0 & 0 \\ 1 & 1 & 1 & \dots & 0 & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ 1 & 1 & 1 & \dots & 1 & 1 \end{bmatrix}. \qquad (14)$$

To simplify our presentation, introduce the operator T to map the vector $h = [h_1, h_2, \cdots, h_N]'$ to a lower triangular Toeplitz matrix H_p, i.e., $H_p = T(h)$. Therefore, we can write

$$T_I = T([1, 1, \cdots, 1]'). \qquad (15)$$

Since $Y_k = H_p U_k$, from (13), we have

$$E_{k+1} = H_e E_k = T(h_e) E_k \qquad (16)$$

where

$$H_e = I - \gamma H_p - k_I H_p T_I. \qquad (17)$$

Denote

$$v_i = [1, 0, \cdots, 0]' \in R^{i\times 1}$$

and

$$h_I = T_I h = [h_{I_1}, h_{I_2}, \cdots, h_{I_N}]$$

with $h_{I_i} = \sum_{j=1}^{i} h_j$. Then, we can write

$$h_e = v_N - [h_I | h][k_I, \gamma]'. \qquad (18)$$

The learning process is governed by (16) and the convergence condition is, by referring to (9), that

$$\|H_e\|_i < 1. \qquad (19)$$

Clearly, if all eigenvalues of H_e, denoted by $\lambda(H_e) = [\lambda_1, \cdots, \lambda_N]'$, are absolutely less than one, the learning process will converge. However, $\max_i |\lambda_i| < 1$ does not imply (19). The consequence is that $\|E_k\|_i$ may not converge monotonically. This is a widely accepted fact. In practice, we are more concerned with the monotonic convergence of 1-norm, ∞-norm and 2-norm of E_k. The convergence conditions are corresponding to replacing 'i' in (19) with '1', '∞' or '2'.

Since H_e is a lower triangular Toeplitz matrix, it is easy to see that

$$\|H_e\|_1 = \|H_e\|_\infty. \qquad (20)$$

Furthermore, $\|H_e\|_1 = \|T(h_e)\|_1 < 1$ if and only if $\|h_e\|_1 < 1$. As shown in [16], if $\|h_e\|_1 < 1$, then $\|H_e\|_2 < 1$. Conversely, if $\|H_e\|_2 < 1$, then $\|h_e\|_2 < 1$. However, it is important to note that $\|H_e\|_2 < 1$ does not imply $\|h_e\|_1 < 1$.

So, the condition $\|h_e\|_1 < 1$ is a sufficient condition for monotonic convergence of 1-norm, ∞-norm and 2-norm of E_k. The ILC design task becomes to optimizing $\|h_e\|_1 < 1$ with respect to k_I and γ. This is possible using numerical simplex method. Here we present a simple method with explicit formulae. We define the following optimization problem for ILC design

$$J^*_{PI} = \min_{k_I,\gamma} J_{PI} = \min_{k_I,\gamma} \|h_e\|_2^2.$$

Since $\|h_e\|_1 < \sqrt{N}\|h_e\|_2$, when J_{PI} is small, it is possible to ensure that $\|h_e\|_1 < 1$.

Let $H = [h_I \vdots h] \in R^{N\times 2}$ and $g = [k_I, \gamma]'$. Then,

$$\begin{aligned} J_{PI} &= [v_N - Hg]'[v_N - Hg] \\ &= 1 - 2v_N' Hg + g'H'Hg. \end{aligned} \qquad (21)$$

The optimal g is simply

$$g^* = (H'H)^{-1} H' v_N \qquad (22)$$

and

$$J^*_{PI} = 1 - v_N' H g^* = h_1(k_I^* + \gamma^*).$$

Using some simple manipulations, (22) can be written in the following form:

$$g^* = \begin{bmatrix} h_I' h_I & -h_I' h \\ -h_I' h & h'h \end{bmatrix}^{-1} \begin{bmatrix} 1 \\ 1 \end{bmatrix} h_1. \qquad (23)$$

Therefore,

$$k_I^* = \frac{h_1(h'h - h_I'h)}{h_I'h_I h'h - (h_I'h)^2}, \qquad (24)$$

$$\gamma^* = \frac{h_1(h_I'h_I - h_I'h)}{h_I'h_I h'h - (h_I'h)^2} \qquad (25)$$

and

$$J^*_{PI} = 1 - h_1^2 \frac{(h'h + h_I'h_I - 2h_I'h)}{h_I'h_I h'h - (h_I'h)^2}. \qquad (26)$$

As a comparison, similar to the above optimization process, we can get the optimal setting of the learning gain for P-type ILC. In this case, $k_I = 0$, $h_e = v_N - \gamma h$. The optimal learning gain to optimize $J_P(\gamma) = \|h_e\|_2$ is given by

$$\gamma^* = h_1/(h'h) \quad (27)$$

and

$$J_P^* = J_P(\gamma^*) = 1 - h_1^2/(h'h). \quad (28)$$

It is expected that for a given h, $J_{PI}^* < J_P^*$. This means that the PI-type ILC can be better than P-type ILC in terms of monotonic convergence speed.

Remark 3.1 *At the first look of (27) and (28), it seems that it is always possible to make $\|E_k\|_2$ converge monotonically if we do not care about the speed of convergence. Since $h'h$ may be quite big for unstable or highly oscillatory system, γ, according to (27), has to be very small and in turn J_P^* is very near to 1 which leads to very slow convergence. We comment that this situation can be alleviated by using a PI-type ILC scheme.*

4 Two Extreme Cases

The expectation that $J_{PI}^* < J_P^*$ is tedious to verify for any vector h which corresponds to the Markov parameters of the plant H_p. Here we offer two extreme cases to show that PI-type ILC is better than P-type ILC, i.e., $J_{PI}^* < J_P^*$ when both of the schemes are optimally designed according to the formulae derived in the previous section.

- **Case 1.** We assume that

$$h = [1, -1, 1, -1, \cdots, 1, -1]'$$

which means that the $\mathcal{Z}$-transfer function of the system is $z/(1+z)$. For convenience, let $N = 2N'$. This is an extreme case for highly oscillatory systems. Then, we have $h_1 = 1$ and

$$h_I = [1, 0, 1, 0, \cdots, 1, 0]'.$$

When P-type ILC is considered, from (27) and (28) , the optimal values are given by

$$\begin{cases} \gamma^* = 1/N \\ J_P^* = 1 - 1/N \end{cases} \quad (29)$$

While for PI-type ILC, by noting that $h'h = N$, $h_I'h_I = N/2 = N'$ and $h_I'h = N/2 = N'$, the optimal values by (24), (25) and (26) are as follows:

$$\begin{cases} \gamma^* = 0 \\ k_I^* = 1/N' = 2/N \\ J_{PI}^* = 1 - 1/N' = 1 - 2/N \end{cases} \quad (30)$$

Clearly, $J_{PI}^* < J_P^*$.

- **Case 2.** h is assumed to be

$$h = [1, 1, 1, 1, \cdots, 1, 1]'$$

which means that the $\mathcal{Z}$-transfer function of the system is $z/(z-1)$. This is an extreme case for very lightly damped systems. Then, we have $h_1 = 1$ and

$$h_I = [1, 2, 3, 4, \cdots, N-1, N]'.$$

Note that here $h'h = N$, $h_I'h_I = N(N+1)(2N+1)/6$ and $h_I'h = N(N+1)/2$. When P-type ILC is considered, the optimal values are the same as given in (29). While for PI-type ILC, the optimal values by (24), (25) and (26), after some algebraic manipulations, are as follows:

$$\begin{cases} \gamma^* = 4/N \\ k_I^* = -6/[N(N+1)] \\ J_{PI}^* = [1 - 3/(N+1)]J_P^* \end{cases} \quad (31)$$

Again, obviously, $J_{PI}^* < J_P^*$ in this case.

Remark 4.1 *From the above computations, we found that the role of N, the length of an iteration, is quite critical in achieving a monotonic convergence. The bigger the N, the more slowly the $\|E_k\|_2$ converges. It has already been discovered in the literature that smaller N makes iterative learning process easier to behave nicely. The above discussion gives another insightful look of this issue.*

Remark 4.2 *We can observe from the two extreme cases that, when N is small, PI-type ILC can be more helpful than P-type ILC in achieving monotonic learning convergence. However, when N is large, $J_{PI}^* \approx J_P^*$. This implies that I-term in ILC updating law is of little help when N is large. Since in practice N is usually large, the I-term is actually not recommended to use. This will also be illustrated in the simulation examples.*

5 Simulation Illustrations

In our simulation examples, second order IIR models will be used. All initial conditions are set to 0. Further, h_1 of the IIR models are all 1. In all simulation experiments, if not otherwise specified, we fix the maximum number of iterations to 40. The desired trajectory of N points is a triangle with a maximum height 1 which is given by

$$y_d(t) = \begin{cases} 2t/N & , \quad i = 1, \cdots, N/2 \\ 2(N-t)/N & , \quad i = N/2+1, \cdots, N. \end{cases} \quad (32)$$

Example 1. Here we use $N = 20$. The $\mathcal{Z}$-transfer function for simulation is

$$H_1(z) = \frac{z - 1.1}{(z + 1.01)(z - 1.01)}. \quad (33)$$

We apply the simple P-type ILC scheme (8) using $\gamma = 1$. The results are summarized in Fig. 1. Fig. 1 shows the P-type ILC results using the learning gain $\gamma = 1$ in updating law (8). Clearly, $\gamma = 1$ is optimal for $h_1 = 1$ which makes $\|1 - \gamma h_1\| = 0$. The top left subplot of Fig. 1 shows the desired trajectory and the output at the 40-th ILC iteration where we can observe the good tracking result. The desired input signal at the 40-th ILC iteration is shown in the top right subplot of Fig. 1. The impulse response of $H_1(z)$ is shown in the bottom right subplot of Fig. 1 where we can read that h_1=1 and $\sum_{j=2}^{40} = 20.6892$. Clearly, the monotonic condition (11) is not satisfied. Therefore, in the bottom left subplot of Fig. 1, a quite big peak transient can be observed for the root mean squares (RMS) of E_k, i.e. RMS(E_k)=$\sqrt{\|E_k\|_2^2/N}$.

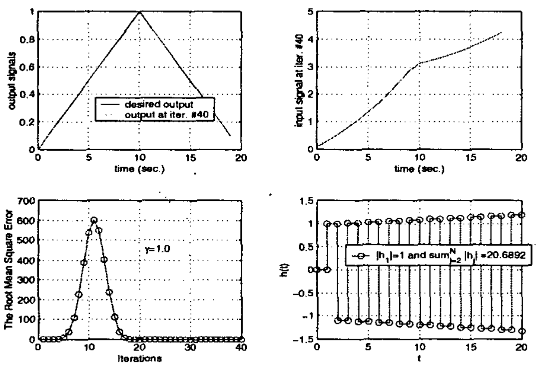


Figure 1: P-type ILC result for slightly unstable highly oscillatory system $H_1(z)$

As discussed in Sec. 3, there exists an optimal choice of γ. In this simulation example, $h_1 = 1$, $h'h = 26.6079$, $h_I'h_I = 8.8952$, and $h'h = 26.6079$. For P-type ILC, the optimal learning gain $\gamma^* = 0.0376$. When PI-type ILC is used, the optimal learning gains are $k_i^* = 0.2930$ and $\gamma^* = -0.1155$. Note that it is somewhat surprised to see the negative value of γ^*. However, in term of $e_k(t+1)$ term, the net gain should be $k_i^* + \gamma^*$ which is still positive. We summarize the results for optimal P and PI ILC schemes in Fig. 2. It is clearly seen that PI scheme performs better than P scheme alone.

Example 2. Consider a stable oscillatory system as follows:

$$H_1(z) = \frac{z - 0.8}{(z - 0.5)(z - 0.9)}. \tag{34}$$

With the simple P-type ILC scheme (8) using $\gamma = 1$ and $N = 60$, the results are summarized in Fig. 3, which is similar to Fig. 1. Again, we observe that the monotonic condition (11) is not satisfied (h_1=1 and $\sum_{j=2}^{40} = 2.995$.)and in the bottom left subplot of Fig. 3, a quite big peak transient in RMS(E_k).

As discussed in Sec. 3, there exists an optimal choice of γ in terms of minimizing $\|h_e\|_2^2$ or $\|H_e\|_2^2$. In this example, $h_1 = 1$, $h'h = 1.7608$, $h_I'h_I = 24.4874$, and $h'h = 1.0797$. For P-type ILC, the optimal learning

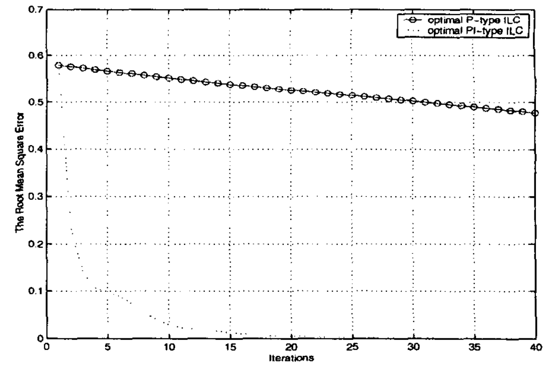


Figure 2: Comparison of optimal P-type and PI-type ILC for $H_1(z)$

gain $\gamma^* = 0.5679$. When PI-type ILC is used, the optimal learning gains are $k_i^* = 0.0162$ and $\gamma^* = 0.5580$. The monotonic convergence can be achieved via either P-type or PI-type ILC scheme using the optimal gains. The RMS(E_k) for this case is shown in Fig. 4(a) where we can see, as expected, that PI-type ILC converges slightly faster than P-type. We understand, as remarked in Remark 4.2, that the contribution from using I-component in ILC updating law becomes less as N increases. To confirm this, let's set N=10. We record the following data: $h_1 = 1$, $h'h = 1.7205$, $h_I'h_I = 4.5063$, and $h'h = 1.0315$; $\gamma^* = 0.5812$ for P-type and $k_i^* = 0.1030$ and $\gamma^* = 0.5195$ for PI-type. For this small duration case, the convergence history comparison similar to Fig. 4(a) is shown in Fig. 4(b). We can see, by comparing Fig. 4(a) and Fig. 4(b), that the I-component contributed more to the rate of monotonic convergence when N is smaller. This is in accordance with our analysis in previous sections.

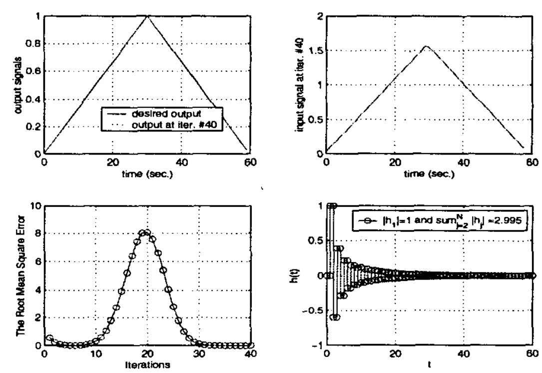
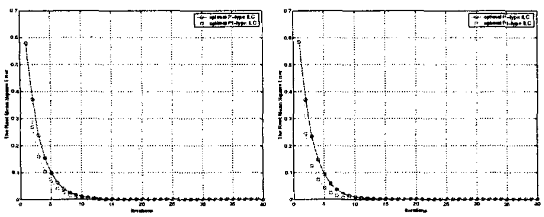


Figure 3: P-type ILC result for stable oscillatory system $H_2(z)$

6 Conclusions

In this paper, we have revisited the PI-type discrete-time iterative learning control scheme with a focus on



(a) $N = 60$ (b) $N = 10$

Figure 4: Comparison of optimal learning performance

the monotonic convergence of the learning process. An optimal design method for PI-type ILC scheme is presented. We have answered this question: *what's the use of the error integral in ILC updating law?* Through analysis and illustrations, both simulation and extreme cases studies, we show that, when the number of time instants in an iteration N is small, the role of the tracking error integral term (I-component) in ILC updating scheme is helpful in achieving a monotonic convergence in a suitable norm topology other than the exponentially weighted sup-norm. However, when N is large, I-component in ILC updating law is practically of little help. Therefore, we do not recommend to use I-term in ILC updating law in practice where N is usually big.

References

[1] S. Arimoto, S. Kawamura, and F. Miyazaki, "Bettering operation of robots by learning," *J. of Robotic Systems*, vol. 1, no. 2, pp. 123–140, 1984.

[2] K. L. Moore, *Iterative learning control for deterministic systems*, Advances in Industrial Control. Springer-Verlag, 1993.

[3] M. Uchiyama, "Formulation of high-speed motion pattern of a mechanical arm by trial," *Trans. SICE (Soc. Instrum. Contr. Eng.)*, vol. 14, no. 6, pp. 706–712(in Japanese), 1978.

[4] Murray Garden, "Learning control of actuators in control systems," *United States Patent #3555252*, 1967.

[5] YangQuan Chen and Kevin L. Moore, "Comments on US Patent #3555252: Learning control of actuators in control systems," in *Proceedings of ICARCV'2000 (The Sixth International Conference on Control, Automation, Robotics and Vision*, Singapore, 2000, pp. CD–ROM.

[6] K. L. Moore, "Iterative learning control - an expository overview," *Applied & Computational Controls, Signal Processing, and Circuits*, vol. 1, no. 1, pp. 151–241, 1999.

[7] Zeungnam Bien and Jian-Xin Xu, *Iterative Learning Control - Analysis, Design, Integration and Applications*, Kluwer Academic Publishers, 1998.

[8] YangQuan Chen and Changyun Wen, *Iterative Learning Control: Convergence, Robustness and Applications*, vol. LNCIS-248 of *Lecture Notes series on Control and Information Science*, Springer-Verlag, London, 1999.

[9] H.-S. Lee and Z. Bien, "Study on robustness of iterative learning control with non-zero initial error," *Int. J. of Control*, vol. 64, no. 3, pp. 345–359, 1996.

[10] Hak-Sung Lee and Zeungnam Bien, "A note on convergence property of iterative learning controller with respect to sup norm," *Automatica*, vol. 33, no. 8, pp. 1591–1593, 1997.

[11] D. H. Owens, "Iterative learning control - convergence using high gain feedback," in *Proc. of the 31st Conf. on Decision and Control*, Tucson, Arizona, USA, Dec. 1992, pp. 2545–2546.

[12] P. Lucibello, "On the role of high gain feedback in P-type learning control of robots," in *Proc. of the 32nd IEEE Conf. on Decision and Control*, San Antonio, Texas, USA, Dec. 1993, pp. 2149–2152.

[13] H. S. Jang and R. W. Longman, "A new learning control law with monotonic decay of the tracking error norm," in *Proceedings of the Fifteenth Annual Allerton Conference on Communication, Control and Computing*, Coordinated Sciences Laboratory, University of Illinois at Urbana-Champaign, 1994, pp. 314–323.

[14] H. Elci, R. W. Longman, M. Phan, J. N. Juang, and R. Ugoletti, "Discrete frequency based learning control for precision motion control," in *Proceedings of the 1994 IEEE International Conference on Systems, Man, and Cybernetics*, San Antonio, TX, 1994, vol. 3, pp. 2767–2773.

[15] Kevin L. Moore, "An observation about monotonic convergence in discrete-time, P-type iterative learning control," in *Proceedings of IEEE Int. Symposium on ISIC'01*, MX, USA, 2001, pp. CD–ROM.

[16] T. Ishihara, K. Abe, and H. Takeda, "A discrete-time design of robust iterative learning controllers," *IEEE Trans. on Systems, Man and Cybernetics*, vol. 22, no. 1, pp. 74–84, 1992.

[17] S. Hillenbrand and M. Pandit, "An iterative learning control with reduced sampling rate for plants with variations of initial states," *Int. J. of Control*, vol. 73, no. 10, pp. 882–889, 2000.

[18] A. Tayebi and M. B. Zaremba, "Exponential convergence of an iterative learning controller for time varying nonlinear systems," in *Proc. of the 38th IEEE Conference on Decision and Control*, Phoenix, Arizona USA, Dec. 1999, IEEE, pp. 1593–1598.

[19] Kevin L. Moore, YangQuan Chen, and Vikas Bahl, "Monotonically convergent iterative learning control for linear discrete-time systems," *for submission to Automatica*, pp. 1–16, July 2001.

[20] Kevin L. Moore, "Multi-loop control approach to designing iterative learning controllers," in *Proceedings of the 37th IEEE Conference on Decision and Control*, Tampa, Florida, USA, 1998, pp. 666–671.