Proceedings of the 2002 IEEE
International Symposium on Intelligent Control
Vancouver, Canada • October 27-30, 2002

An Optimal Design of PD-type Iterative Learning Control with Monotonic Convergence [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 technique to make use of the repetitiveness of the tasks a system is commanded to execute in a fixed finite time interval. In this paper, we assume that a measured finite impulse response series of the plant to be controlled is available. We present an optimal design procedure for the commonly used PD-type ILC updating law. Monotonic convergence in a suitable norm topology other than the exponentially weighted sup-norm is emphasized. For practical reasons, an averaged difference formula for a numerical derivative estimate is preferred over the conventional one step backward difference method for smoothing out the high frequency noise. Via analysis, we show a trade-off between noise suppression and rate of monotonic convergence of the ILC process.

Keywords: Iterative learning control; monotonic convergence; proportional plus derivative (PD) type ILC, optimal design, numerical derivative estimate.

1 Introduction

We know that when we do the same job over and over we can build our "*skill*" for this specific job. If a machine is commanded to execute the same task repeatedly, should we expect the machine to perform better and better? Intuitively, the answer is "*yes*". How to achieve this is through "*iterative learning control*," which is popular because of many applications where the task is to be executed repeatedly, such as batch chemical reactors, robot manipulators, wafer steppers, hard disk servo controllers etc. Usually, in these repetitive systems, the feedback controller design did *not* take into account this repetitiveness. By making use of this repetitiveness, there is a room for further performance improvement. Iterative learning control (ILC) [1, 2] is a technique 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].

Early research efforts on ILC schemes mainly considered their *convergence analysis*, without explicit design or synthesis procedures. However, the convergence conditions found in the literature are typically not sufficient for actual ILC applications. Therefore, in recent years increasing efforts have been made on the *design* issue of ILC. These can be observed from the latest books [7, 8] and the dedicated ILC web server [9]. A recent survey on the ILC *design* issue [10] has documented various practically tested design schemes, though mainly for robotic manipulators. However, to draw attention from industry, the existing design techniques are still not sufficiently attractive as compared to the successful use of PID (proportional-integral-derivative) controllers in industry. The ILC design issue should better be attacked in a way similar to PI/PID controller design. With less plant model information, some design methods similar to PI/PID controller parameter setting have been proposed, for example, a three-parameter design procedure using closed-loop Bode plots [10], a local-symmetric-integral (LSI) type two-parameter design method [11, 12], a two-parameter design method using learning feedforward control (LFFC) with B-splines networks (BSN) [13, 14, 15, 16], a frequency domain adaptive LFFC [17], etc. The design method for ILC applications depends on what kind of plant model information assumed. Developing different design procedures with different kinds of assumed plant knowledge will be a continuing effort in ILC research.

[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. Project website: `http://www.csois.usu.edu`. Corresponding author: Dr YangQuan Chen. E-mail: `yqchen@ieee.org`; Tel. 01-435-7970148; Fax: 01-435-7972003. URL: `http://www.crosswinds.net/~yqchen`.

We would like to argue that, first of all, ILC design objectives should be first focused on the monotonic convergence issue. Monotonic convergence means "*better and better*" from trial to trial. In most practical situations, "*better-worse-better*" is not allowable. It is observed in [18] that although the λ-norm of the 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 decaying learning gain as discussed in [18]. One may argue that, to make the convergence monotonic in the up-norm or the 2-norm, one can use a high-gain feedback [19, 20]. 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 [21, 22] 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 the time domain, a condition for monotonic convergence of ∞-norm of tracking errors is established in [23]. There are some analysis results for monotonic convergence of ILC schemes via using approximate impulse response [24], reduced sampling rate [25] and for sampled data nonlinear systems [26]. How to achieve the monotonic convergence for discrete time systems via a proper ILC updating law design is addressed in a recent work [27] where a time-varying learning gain is used to achieve monotonic learning convergence. Furthermore, in [28], via extensive simulation experiments and mathematical analysis, two important facts were presented: **(1)** the use of high-order controller dynamics in the time-axis is to condition the system dynamics so that a monotonic convergence can be achieved and **(2)** the use of high-order dynamics in the iteration-axis is to reject the iteration-dependent disturbance by virtue of the internal model principle (IMP). So, for monotonic convergence, high-order ILC in the iteration direction [29, 30, 31] may not be really helpful, as also recently discussed in [32].

Based on above discussions, we focus in this paper on the simple PD-type ILC scheme with only two tuning knobs. We concentrate on the first-order ILC using the information of the previous iteration only. In many applications, via experiments, the finite impulse response series is available. Based on the experimental finite impulse response series, we present an optimal parameter setting method for the two tuning parameters. To suppress the high frequency noise in the tracking error measurements, an averaged difference formula for the numerical derivative estimate is preferred over the conventional one step backward difference method. Via analysis, we show that there is a trade-off between noise suppression and rate of monotonic convergence of ILC process.

The rest of this paper is organized as follows. In Sec. 2, some notations and preliminaries are given. In Sec. 3, the monotonic convergence condition for the PD-type ILC scheme is analyzed together with an optimal design method. In Sec. 4, the averaged difference formula for numerical derivative estimate is used and its effect on the ILC convergence rate is analyzed. Finally, conclusions are presented in Sec. 5 with several remarks on the further investigations.

2 Notations and Preliminaries

In what follows, the subscript "k" denotes the k-th iteration (trial, repetition). The discrete time index t in a given trial ranges from 0 to N, i.e., $t \in [0, N]$. 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):

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

where z^{-1} is the standard delay operator in time, and the parameters h_i are the standard Markov parameters of the system $H(z)$. Note that we assume with no loss of generality the relative degree of the system is one in the sequel. 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" [33]

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

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

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

and

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

then the system can be written as

$$Y_k = H_p U_k, \tag{2}$$

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

$$H_p \triangleq \begin{bmatrix} h_1 & 0 & 0 & \dots & 0 \\ h_2 & h_1 & 0 & \dots & 0 \\ h_3 & h_2 & h_1 & \dots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ h_N & h_{N-1} & h_{N-2} & \dots & h_1 \end{bmatrix}. \tag{3}$$

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

$$u_{k+1}(t) = u_k(t) + \gamma e_k(t+1) \tag{4}$$

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.

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

$$\|I - \gamma H_p\|_i < 1. \tag{5}$$

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 [33]:

$$|1 - \gamma h_1| < 1. \tag{6}$$

Unfortunately, this second condition does *not* guarantee monotone convergence as observed in [23]. In addition to the necessary and sufficient condition for convergence (6), the other conditions to guarantee monotone convergence can be found in a theorem given in [23], i.e.,

$$|h_1| > \sum_{j=2}^{N} |h_j|. \tag{7}$$

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 (5) using optimization techniques based on the available tuning knobs, e.g., γ in the P-type scheme (4). In a PD-type scheme, an extra tuning knob is introduced.

3 Generic PD-type ILC and Its Optimal Design

By using a one step backward finite difference as the approximation of the derivative (D) signal, the PD-type ILC is given by

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

where k_p and k_d are proportional and derivative learning gains respectively. Introduce the operator T to map the column vector $h = [h_1, h_2, \cdots, h_N]'$ to a lower triangular Toeplitz matrix H_p, i.e., $H_p \triangleq T(h)$. For example, let $c_2 = [0, 1, 0, \cdots, 0]'$. Then, we have

$$T_2 \triangleq T(c_2) = \begin{bmatrix} 0 & 0 & 0 & \ldots & 0 & 0 \\ 1 & 0 & 0 & \ldots & 0 & 0 \\ 0 & 1 & 0 & \ldots & 0 & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & 0 & \ldots & 1 & 0 \end{bmatrix}. \tag{9}$$

In the sequel, we shall use a more general notion T_i similar to the definition of T_2. Clearly, for $i = 1$, $T_i = I_N$. Using supervector representation, we can write

$$U_{k+1} = U_k(t) + k_p T_2 E_k + k_d(I_N - T_2)E_k \tag{10}$$

where $I_N = T_1$ is a square identity matrix of dimension N. Since $Y_k = H_p U_k$ and $E_k = Y_d - Y_k$, from (10) we have

$$E_{k+1} = H_e E_k = T(h_e)E_k \tag{11}$$

where

$$H_e = I_N - (k_p - k_d)H_p T_2 - k_d H_p \tag{12}$$

and

$$h_e = v_N - [\bar{h}_2, h - \bar{h}_2][k_p, k_d]'. \tag{13}$$

In the above equation, we used the following notations:

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

and

$$\bar{h}_2 \triangleq T_2 h = [0, h_1, h_2, \cdots, h_{N-1}]'.$$

The learning process is governed by (11) and the convergence condition is, analogous to (5), that

$$\|H_e\|_i < 1. \tag{14}$$

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 (14). The consequence is that $\|E_k\|_i$ may not converge monotonically, which is widely recognized. In practice, we are more concerned with the monotonic convergence of the 1-norm, ∞-norm and 2-norm of E_k. The convergence conditions are corresponding to replacing 'i' in (14) with '1', '∞' or '2'.

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

$$\|H_e\|_1 = \|H_e\|_\infty. \tag{15}$$

Furthermore, $\|H_e\|_1 = \|T(h_e)\|_1 < 1$ if and only if $\|h_e\|_1 < 1$. As shown in [24], 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 the 1-norm, ∞-norm and 2-norm of E_k. The ILC design task becomes to optimizing $\|h_e\|_1 < 1$ with respect to k_p and k_d. This is possible using a numerical simplex method. Here we present a simple method with explicit formulae. We define the following optimization problem for ILC design

$$J^*_{PD} = \min_{k_p, k_d} J_{PD} \triangleq \min_{k_p, k_d} \|h_e\|_2^2.$$

Since $\|h_e\|_1 < \sqrt{N}\|h_e\|_2$, when J^*_{PD} is small, it is possible to ensure that $\|h_e\|_1 < 1$.

Let $H = [\bar{h}_2, h - \bar{h}_2] \in R^{N\times 2}$ and $g = [k_p, k_d]'$. Then,

$$\begin{aligned} J_{PD} &= [v_N - Hg]'[v_N - Hg] \\ &= 1 - 2v_N' Hg + g'H'Hg. \end{aligned} \quad (16)$$

The optimal g is simply

$$g^* = [k_p^*, k_d^*]' = (H'H)^{-1}H'v_N \quad (17)$$

and

$$J_{PD}^* = 1 - v_N' Hg^* = 1 - h_1 k_d^*. \quad (18)$$

After some simple manipulations, (17) can be written in the following form:

$$g^* = \begin{bmatrix} \bar{h}_2'\bar{h}_2 & \bar{h}_2'(h-\bar{h}_2) \\ \bar{h}_2'(h-\bar{h}_2) & (h-\bar{h}_2)'(h-\bar{h}_2) \end{bmatrix}^{-1} \begin{bmatrix} 0 \\ h_1 \end{bmatrix}. \quad (19)$$

Therefore,

$$k_p^* = -\frac{h_1 \bar{h}_2'(h-\bar{h}_2)}{\bar{h}_2'\bar{h}_2(h-\bar{h}_2)'(h-\bar{h}_2) - [\bar{h}_2'(h-\bar{h}_2)]^2}, \quad (20)$$

$$k_d^* = \frac{h_1 \bar{h}_2'\bar{h}_2}{\bar{h}_2'\bar{h}_2(h-\bar{h}_2)'(h-\bar{h}_2) - [\bar{h}_2'(h-\bar{h}_2)]^2} \quad (21)$$

and

$$J_{PD}^* = 1 - \frac{h_1^2 \bar{h}_2'\bar{h}_2}{\bar{h}_2'\bar{h}_2(h-\bar{h}_2)'(h-\bar{h}_2) - [\bar{h}_2'(h-\bar{h}_2)]^2}. \quad (22)$$

Remark 3.1 *It is interesting to observe from (18) that $J_{PD}^* = 1$ when $k_d = 0$ in (8) with a simpler ILC updating law*

$$u_{k+1}(t) = u_k(t) + k_p e_k(t). \quad (23)$$

Note that, in this case, we cannot expect monotonic convergence of ILC since $J_{PD}^* = 1$. This in turn verifies that a correct time advance step, which corresponds to the system relative degree, such as the form in (4) is essential.

As a comparison, similar to the above optimization process, we can get the optimal setting of the learning gain for the P-type ILC given in (4) which is a special case of (8) when $k_d = k_p = \gamma$. In this case, correspondingly, we have $h_e = v_N - \gamma h$. The optimal learning gain to optimize $J_P(\gamma) = \|h_e\|_2$ is given by

$$\gamma^* = h_1/(h'h) \quad (24)$$

and

$$J_P^* = J_P(\gamma^*) = 1 - h_1^2/(h'h). \quad (25)$$

Remark 3.2 *At the first look of (24) and (25), 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 (24), has to be very small and in turn J_P^* is very near to 1 which leads to a very slow convergence. We comment that this situation can be alleviated by using a PD-type ILC scheme.*

It is expected that for a given nominally measured h, $J_{PD}^* < J_P^*$. This means that the optimally designed PD-type ILC can be better than the optimally designed P-type ILC in terms of monotonic convergence speed. It is tedious to verify for any vector h which corresponds to the Markov parameters of the plant H_p. Let's examine two simple extreme cases.

Case 1. Let $h = [1, -1, 1, -1, \cdots, 1, -1]'$, i.e., the system is $z/(1+z)$ which is an extreme case for highly oscillatory systems. When the P-type ILC (4) is considered, the optimal values from (24) and (25) are $\gamma^* = 1/N$ and $J_P^* = (N-1)/N$. With a PD-type ILC (8), the optimal values via (20), (21) and (22) are $k_p^* = 2$, $k_d^* = 1$ and $J_{PD}^* = 0$. Clearly, $J_{PD}^* < J_P^*$.

Case 2. Let $h = [1, 1, 1, 1, \cdots, 1, 1]'$, i.e., the system is $z/(-1+z)$ which is an extreme case for very lightly damped systems. For the P-type ILC (4), the optimal values are the same as in **Case 1.** With a PD-type ILC (8), the optimal values are $k_p^* = 0$, $k_d^* = 1$ and $J_{PD}^* = 0$. Again, $J_{PD}^* < J_P^*$.

Remark 3.3 *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 for P-type ILC (4). This explains why the P-type ILC converges slowly especially when N is large.*

Remark 3.4 *We remark that, in the literature, the following discrete-time ILC*

$$u_{k+1}(t) = u_k(t) + k_d(e_k(t+1) - e_k(t)) \quad (26)$$

is frequently used as a counterpart of the classical Arimoto D-type ILC in continuous time domain. According to **Case 1**, *the optimal ILC law is actually given by*

$$u_{k+1}(t) = u_k(t) + k_d(e_k(t+1) + e_k(t)). \quad (27)$$

We argue that, from monotonic convergence point of view, we should not be confined to (26).

4 Optimal PD-type ILC Design Using Averaged Derivative Formula

In the generic PD-type ILC scheme (8), a simple one step backward finite difference is used to estimate the derivative signal. For a better noise suppression, it is a common practice to use a central difference formula. In this case, (8) becomes

$$u_{k+1}(t) = u_k(t) + k_p e_k(t) + k_d(e_k(t+1) - e_k(t-1))/2. \quad (28)$$

The derivative estimate $(e_k(t+1) - e_k(t-1))/2$ can be regarded as an averaged value from two derivative estimates $e_k(t+1) - e_k(t)$ and $e_k(t) - e_k(t-1)$. For a more general averaged formula, we consider the following PD-type ILC scheme

$$u_{k+1}(t) = u_k(t) + k_p e_k(t) + \frac{k_d}{m}(e_k(t+1) - e_k(t-m+1)) \tag{29}$$

where $m > 0$ is the number of averaging points. Clearly, (8) is a special case of (29) when $m = 1$. The value of m depends on the noise suppression requirement. In practice, m can be chosen between 1 to 4.

Based on (29), the optimal design of k_p and k_d is similar to the procedures developed in Sec. 3. Starting from (11), using (29), we now have

$$H_e = I_N - k_p H_p T_2 - k_d H_p/m + k_d H_p T_m/m \tag{30}$$

and

$$h_e = v_N - [\bar{h}_2, (h - \hat{h}_m)/m][k_p, k_d]' \tag{31}$$

where $\hat{h}_m = [0_{1\times m}, h_1, h_2, \cdot, h_{N-m}]'$. Similarly, we can get

$$g^* = \begin{bmatrix} \bar{h}_2'\bar{h}_2 & \bar{h}_2'(h - \hat{h}_m)/m \\ \frac{\bar{h}_2'(h-\hat{h}_m)}{m} & \frac{(h-\hat{h}_m)'(h-\hat{h}_m)}{m^2} \end{bmatrix}^{-1} \begin{bmatrix} 0 \\ \frac{h_1}{m} \end{bmatrix}. \tag{32}$$

Therefore,

$$k_p^* = -\frac{h_1\bar{h}_2'(h - \hat{h}_m)}{\bar{h}_2'\bar{h}_2(h - \hat{h}_m)'(h - \hat{h}_m) - [\bar{h}_2'(h - \hat{h}_m)]^2}, \tag{33}$$

$$k_d^* = \frac{m h_1 \bar{h}_2'\bar{h}_2}{\bar{h}_2'\bar{h}_2(h - \hat{h}_m)'(h - \hat{h}_m) - [\bar{h}_2'(h - \hat{h}_m)]^2} \tag{34}$$

and from $J_{PD}^* = 1 - [0, h_1/m]g^*$,

$$J_{PD}^* = 1 - \frac{h_1^2\bar{h}_2'\bar{h}_2}{\bar{h}_2'\bar{h}_2(h - \hat{h}_m)'(h - \hat{h}_m) - [\bar{h}_2'(h - \hat{h}_m)]^2}. \tag{35}$$

To show the trade-off between the noise suppression and the rate of monotonic convergence of ILC process, re-consider the two extreme cases in Sec. 3. Here we consider $m = 2$. For **Case 1** , the optimal values via (33), (34) and (35) are $k_p^* = 1/(2N-3)$, $k_d^* = (2N-2)/(2N-3)$ and $J_{PD}^* = (N-2)/(2N-3)$. For **Case 2.** $k_p^* = -1/(2N-3)$; k_d^* and J_{PD}^* are the same as **Case 1**. Recall that J_{PD}^* in Sec. 3 is 0. Clearly, the smoothing or averaging scheme for noise suppression is at the expense of slowing down the best achievable ILC monotonic convergence rate. This trade-off should be taken into account during ILC applications.

5 Conclusions

In this paper, we have presented an optimal design procedure for the commonly used PD-type ILC updating law. Monotonic convergence in a suitable norm topology other than the exponentially weighted sup-norm is emphasized. For practical reason, an averaged difference formula for numerical derivative estimate is preferred over the conventional one step backward difference method in smoothing out the high frequency noise. Via analysis, we show a trade-off between the noise suppression and the rate of monotonic convergence of ILC process.

In our further research efforts, the uncertainties in the measured impulse response function will be addressed involving the norm minimization of an interval Toeplitz matrix. Moreover, J_{PD}^* achieved in this paper can be further optimized by introducing additional tuning knobs. We remark that the optimal PD-type ILC using a time-varying learning gain will be an interesting problem.

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 *Proc. of ICARCV'2000 (The Sixth International Conference on Control, Automation, Robotics and Vision*, Singapore, 2000.

[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] YangQuan Chen, "Dedicated web server for Iterative Learning Control research," `http://cicserver.ee.nus.edu.sg/~ilc`, Oct. 1998.

[10] Richard W. Longman, "Designing iterative learning and repetitive controllers," in *Iterative Learning Control - Analysis, Design, Integration and Application*, Zeungnam Bien and Xu Jian-Xin, Eds., chapter 6, pp. 107–145. Kluwer Academic Publishers, 1998.

[11] Y. Q. Chen, H. F. Dou, and K. K. Tan, "Iterative learning control via weighted local-symmetrical-integration," *Asian Journal of Control*, vol. 3, no. 4, pp. 352-356, 2001.

[12] Y. Q. Chen and K. L. Moore, "A practical iterative learning path-following control of an omni-directional vehicle," *Asian Journal of Control* (Special Issue on Iterative Learning Control), vol. 4, no. 1, pp. 90-98, 2002.

[13] T. J. A. de Vries, W. J. R. Velthuis, and J. van Amerongen, "Learning feed-forward control: A survey and historical note," in *Proc. of the First IFAC Conference on Mechatronic Systems*, Darmstadt, Germany, 2000, pp. 949 –954.

[14] W. J. R. Velthuis, T. J. A. de Vries, and J. van Amerongen, "Design procedure for a learning feed-forward controller," in *Proc. of the First IFAC Conference on Mechatronic Systems*, Darmstadt, Germany, 2000, pp. 955 –960.

[15] YangQuan Chen and Kevin L. Moore, "Frequency domain analysis and design of learning feed-forward controller using the second order B-spline network," *Revised for Automatica*, Dec. 2001.

[16] YangQuan Chen, Kevin L. Moore, and Vikas Bahl, "Improved path following of USU ODIS by learning feedforward controller using dilated B-spline network," in *CDROM Proc. of The 2001 IEEE International Symposium on Computational Intelligence in Robotics and Automation (IEEE CIRA 2001)*, Banff, Alberta, Canada, 2001, IEEE, pp. 59-64.

[17] YangQuan Chen and Kevin L. Moore, "Frequency domain adaptive learning feedforward control," in *CDROM Proc. of The 2001 IEEE International Symposium on Computational Intelligence in Robotics and Automation (IEEE CIRA 2001)*, Banff, Alberta, Canada, 2001, IEEE, pp. 396-401.

[18] 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.

[19] 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.

[20] 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.

[21] 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.

[22] 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.

[23] Kevin L. Moore, "An observation about monotonic convergence in discrete-time, P-type iterative learning control," in *Proceedings of IEEE Int. Symposium on Intelligent Control (ISIC'01)*, MX, USA, 2001.

[24] 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.

[25] 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.

[26] 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, pp. 1593–1598.

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

[28] K. L. Moore and YangQuan Chen, "On the monotonic convergence of high order iterative learning update laws," Proceedings of The 15-th IFAC Congress, Barcelona, Spain, July 21-26 2002, IFAC. Accepted and to be presented.

[29] Z. Bien and K. M. Huh, "High-order iterative learning control algorithm," *IEE Proceedings, Part-D, Control Theory and Applications*, vol. 136, no. 3, pp. 105–112, 1989.

[30] Y. Chen, Z. Gong, and C. Wen, "Analysis of a high order iterative learning control algorithm for uncertain nonlinear systems with state delay," *Automatica*, vol. 34, no. 3, pp. 345–353, March 1998.

[31] Y. Chen, C. Wen, J.-X. Xu, and M. Sun, "High-order iterative learning identification of projectile's aerodynamic drag coefficient curve from radar measured velocity data," *IEEE Transactions on Control Systems Technology*, vol. 6, no. 4, pp. 563–570, 1998.

[32] M. Norrlöf and S. Gunnarsson, "Comparative study on first and second order ILC - frequency domain analysis and experiments," in *Proc. of the 39th IEEE Conference on Decision and Control*, Sydney, Australia, Dec. 2000.

[33] 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.