

Chapter 9

Iterative Learning Control

Kevin L. Moore

ABSTRACT In this paper we give an overview of the field of iterative learning control (ILC). We begin with a detailed description of the ILC technique, followed by two illustrative examples that give a flavor of the nature of ILC algorithms and their performance. This is followed by a topical classification of some of the literature of ILC and a discussion of the connection between ILC and other common control paradigms, including conventional feedback control, optimal control, adaptive control, and intelligent control. Next, we give a summary of the major algorithms, results, and applications of ILC given in the literature. This discussion also considers some emerging research topics in ILC. As an example of some of the new directions in ILC theory, we present some of our recent results that show how ILC can be used to force a desired periodic motion in an initially non-repetitive process: a gas-metal arc welding system. The paper concludes with summary comments on the past, present, and future of ILC.

9.1 Introduction

Problems in control system design may be broadly classified into two categories: stabilization and performance. In the latter category a typical problem is to force the output response of a (dynamical) system to follow a desired trajectory as close as possible, where “close” is typically defined relative to a specific norm or some other measure of optimality. Although control theory provides numerous tools for attacking such problems, it is not always possible to achieve a desired set of performance design requirements. This may be due to the presence of unmodelled dynamics or parametric uncertainties that are exhibited during actual system operation or to the lack of suitable design techniques for a particular class of systems (e.g., there is not a comprehensive theory of linear quadratic optimal design technique for nonlinear systems).

Iterative learning control is a relatively new addition to the control engineer’s toolkit that, for a particular class of problems, can be used to overcome some of the traditional difficulties associated with performance design of control systems. Specifically, iterative learning control, or ILC, is a technique for improving the transient response and tracking performance of processes, machines, equipment, or systems that execute the same tra-

jectory, motion, or operation over and over. The classic example of such a process is a robotic manipulator performing spot welding in a manufacturing assembly line. For instance, such a manipulator might be programmed to wait in its home position until a door panel is moved into place. It then carries out a series of welds at pre-defined locations, after which it returns to its home position until the door panel is removed. The entire process is then repeated. Although robotic operations and manufacturing present obvious examples of situations in which a machine or process must execute a given trajectory over and over, there are numerous other problems that can be viewed from the framework of repetitive operations. In these situations, iterative learning control can be used to improve the system response. The approach is motivated by the observation that if the system controller is fixed and if the system's operating conditions are the same each time it executes, then any errors in the output response will be repeated during each operation. These errors can be recorded during system operation and can then be used to compute modifications to the input signal that will be applied to the system during the next operation, or trial, of the system. In iterative learning control refinements are made to the input signal after each trial until the desired performance level is reached. Research in the field of iterative learning control focuses on the algorithms that are used to update the input signal. Note that in describing the technique of iterative learning control we use the word *iterative* because of the recursive nature of the system operation and we use the word *learning* because of the refinement of the input signal based on past performance in executing a task or trajectory.

In this paper we give an overview of the field of iterative learning control. We begin with a detailed description of the ILC technique, followed by two illustrative examples that give a flavor of the nature of ILC algorithms and their performance. This is followed by a topical classification of some of the literature of ILC and a discussion of the connection between ILC and other common control paradigms, including conventional feedback control, optimal control, adaptive control, and intelligent control. Next, we give a summary of the major algorithms, results, and applications of ILC given in the literature. This discussion also considers some emerging research topics in ILC. As an example of some of the new directions in ILC theory, we present some of our recent results that show how ILC can be used to forced a desired periodic motion in an initially non-repetitive process: a gas-metal arc welding system. The paper concludes with summary comments on the past, present, and future of ILC.

9.2 Generic Description of ILC

The basic idea of iterative learning control is illustrated in Figure 9.1. All the signals shown are assumed to be defined on a finite interval $t \in [0, t_f]$. The subscript k indicates the trial or repetition number. The scheme operates as follows: during the k -th trial an input $u_k(t)$ is applied to the system, producing the output $y_k(t)$. These signals are stored in the memory units until the trial is over, at which time they are processed off-line by the ILC algorithm (actually, it is not always necessary to wait until the end of the trial to do the processing – it depends on the ILC algorithm you are using). Based on the error that is observed between the actual output and the desired output ($e_k(t) = y_d(t) - y_k(t)$), the ILC algorithm computes a modified input signal $u_{k+1}(t)$ that will be stored in memory until the next time the system operates, at which time this new input signal is applied to the system. This new input should be designed so that it will produce a smaller error than the previous input.

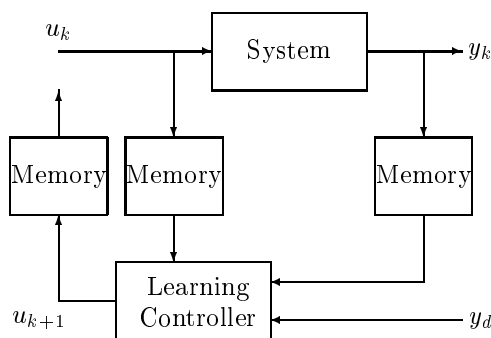


FIGURE 9.1. Iterative learning control configuration.

The iterative learning control approach can be described more precisely by introducing some additional notation. Let the nonlinear operator $f : U \mapsto Y$ mapping elements in the vector space U to those in the vector space Y be written as $y = f(u)$ where $u \in U$ and $y \in Y$. We assume suitably defined norms on U and Y as well as an induced norm on $f(\cdot)$. Suppose we are given a system S , defined by $y(t) = f_S(u(t), t)$ (here we assume $f_S(\cdot, t)$ is an input-output operator and may represent a dynamical system in the usual way). For this system we wish to drive the output to a desired response defined by $y_d(t)$. This is equivalent to finding the optimal input $u^*(t)$ that satisfies

$$\min_{u(t)} \|y_d(t) - f_S(u(t), t)\| = \|y_d(t) - f_S(u^*(t), t)\|.$$

In this context, ILC is an iterative technique for finding $u^*(t)$ for the case in which all the signals are assumed to be defined on the finite interval $[0, t_f]$. The ILC approach is to generate a sequence of inputs, $u_k(t)$ in such a way that the sequence converges to u^* . That is, we seek a sequence

$$\begin{aligned} u_{k+1}(t) &= f_L(u_k(t'), y_k(t'), y_d(t'), t) \\ &= f_L(u_k(t'), f_S(u_k(t')), y_d(t'), t), \quad t' \in [0, t_f], \end{aligned}$$

such that

$$\lim_{k \rightarrow \infty} u_k(t) = u^*(t) \quad \text{for all } t \in [0, t_f].$$

Some remarks about this problem include:

1. In a successful ILC algorithm the next input will be computed so that the performance error will be reduced on the next trial. The issue is usually quantified by saying that the error should converge, with convergence measured in the sense of some norm.
2. It is worth repeating that we have defined our signals with two variables, k and t . The trial is indexed with the integer k , while time is described by the variable t , which may be continuous or discrete.
3. The general algorithm shown introduces a new variable t' . This reflects the fact that after the trial is complete there is effectively no causality restriction in the ILC operator f_L . Thus one may use information about what happened *after* the input $u_k(t_0)$ was applied when constructing the input $u_{k+1}(t_0)$. The only place this is not possible is at $t = t_f$. Although we can assume $t' \in [0, t_f]$, realistically we only need $t' \in [t, t_f]$ when computing $u_{k+1}(t)$.
4. The previous remark is emphasized in Figure 9.2, which illustrates the distinction between conventional feedback and iterative learning control. In Figure 9.2(a) it is shown how the ILC approach preserves information about the effect of the input at each instant during the iteration and uses that information to compute corrections to the control signal during the next trial. Figure 9.2(b) shows that in a conventional feedback strategy the error from the current time step is used by the controller to compute the input for the next time step. However, the effect of this decision is not preserved from one trial to the next.
5. It is usually assumed implicitly that the initial conditions of the system are reset at the beginning of each trial (iteration) to the same value. This has always been a key assumption in the formulation of the ILC problem.
6. It is also usually assumed by definition that the trial length t_f is fixed. Note, however, that it is possible to allow $t_f \rightarrow \infty$. This is often done for analysis purposes.

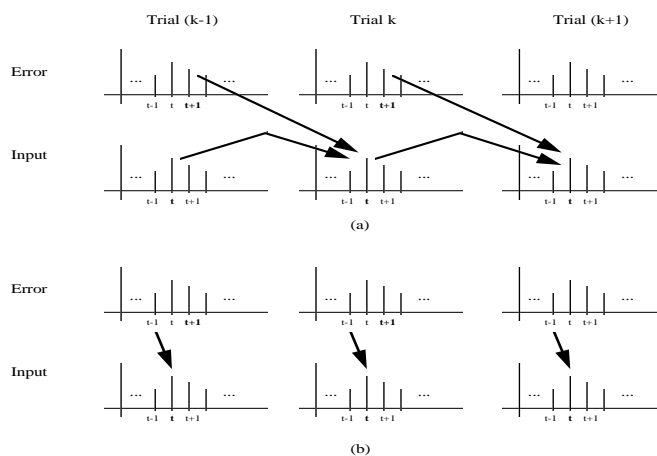


FIGURE 9.2. (a) Iterative learning control; (b) conventional feedback control.

7. Ideally, as little information as possible about the system f_S should be used in designing the learning controller system f_L (at the least it should require minimal knowledge of the system parameters).
8. Notwithstanding the previous comment, in the case of full plant knowledge, the problem is solved if the operator f_S is left invertible and time-invariant. In this case simply let $u^*(t) = f_S^{-1}y_d(t)$. If the system is not invertible and $y_d(t)$ is not in the image of f_S then the best we can hope for is to find a $u^*(t)$ that minimizes $\|y_d(t) - y(t)\|$ over all possible inputs $u(t)$. The goal of the learning control algorithm is to iteratively find such a $u^*(t)$.
9. Ideally, the convergence properties of the ILC algorithm should not depend on the desired response $y_d(t)$. If a new desired trajectory is introduced, the learning controller would simply “learn” the new optimal input, without changing any of its own algorithms.
10. In order to show convergence and to ensure stability in actual implementations, the system S is usually required to be stable. It has been

noted in the literature that if we initially have an unstable plant, then we should first stabilize it using conventional techniques. As we discuss later in the paper, recent research has considered simultaneous use of conventional feedback with iterative learning control. However, the primary role of ILC is to improve performance rather than to stabilize. As such it is clear that the outcome of an ILC algorithm is derivation of the best possible feedforward signal to input to the system in order to achieve a desired output. In this sense the ILC technique produces the output of an optimal prefilter for the system relative to a given desired output and thus can be considered a feedforward design technique.

9.3 Two Illustrative Examples of ILC Algorithms

Before discussing the literature of iterative learning control it is probably helpful to consider some examples of how the ILC algorithms work. First we will look at a simple linear example. Then we will present a more complicated example of an ILC algorithm for motion control of a robotic manipulator.

9.3.1 A Linear Example

Consider the following second-order, discrete-time linear system described by

$$\begin{aligned}y(t+1) &= -.7y(t) - .012y(t-1) + u(t) \\y(0) &= 2, \\y(1) &= 2,\end{aligned}$$

This system is stable but always exhibits an “underdamped” behavior by virtue of the fact that its poles lie on the negative real axis. Suppose we wish to force the system to follow a signal y_d such as that shown in Figure 9.3. This might correspond to a motor velocity trajectory, for example, where the speed is required to step up and then back down. To apply the iterative learning control approach to this system we begin by defining $u_0(t) = y_d(t)$. Because we assume no knowledge of the system we wish to control this is a reasonable definition. Figure 9.4(a) shows the output signal, denoted $y_0(t)$, that results when this input is applied to the system. (for reference the figure also shows the desired output). From this we compute $e_0(t) = y_d(t) - y_0(t)$ and a new input signal $u_1(t)$ is computed according to

$$u_1(t) = u_0(t) + .5e_0(t+1)$$

for $t \in [0, t_f - 1]$. At the endpoint, where $t = t_f$, we use

$$u_1(t_f) = u_0(t) + .5e_0(t_f).$$

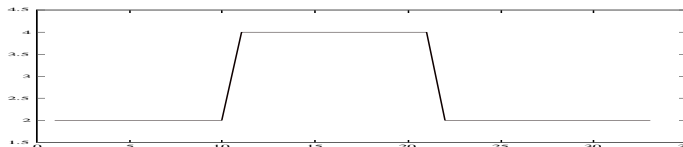


FIGURE 9.3. Desired system response.

This signal is then applied to the system (which has had its initial conditions reset to the same values as during the first trial) and the new output is recorded, a new error is determined, and the next input signal $u_2(t)$ is computed. The process is then repeated until the output converges to the desired signal. Figures 9.4(b) and 4(c) show the output signal after five and ten trials, respectively. It is clear that the algorithm has forced the output to the required value for each instant in the interval. Note also the resulting input signal, $u_{10}(t)$, shown in Figure 9.4(d). It can be seen that the learning control algorithm has derived an input signal that anticipates the dynamics of the process, including the two-step deadtime, the oscillatory poles, and the non-unity DC-gain. Of course, it should be pointed out that in this example we have set the initial conditions equal to the desired output. Certainly one cannot affect the values of the output that occur before an input is applied. However, for linear systems, it can be shown that the algorithm will converge for all time after the deadtime.

9.3.2 An Adaptive ILC Algorithm for a Robotic Manipulator

Because applications of robotic manipulators usually involve some type of repetitive motion it is natural to consider iterative learning control to improve the performance of manipulators. In this section we present an example of an adaptive algorithm for ILC of a robotic manipulator developed by this author (see [160], for example). Consider the simple two-link manipulator shown in Figure 9.5. The manipulator can be modelled by

$$A(x_k)\ddot{x}_k + B(x_k, \dot{x}_k)\dot{x}_k + C(x_k) = u_k,$$

where $u_k(t)$ is a vector of torques applied to the joints and

$$\begin{aligned} x(t) &= (\theta_1(t), \theta_2(t))^T \\ A(x) &= \begin{pmatrix} .54 + .27 \cos \theta_2 & .135 + .135 \cos \theta_2 \\ .135 + .135 \cos \theta_2 & .135 \end{pmatrix}, \\ B(x, \dot{x}) &= \begin{pmatrix} .135 \sin \theta_2 & 0 \\ -.27 \sin \theta_2 & -.135(\sin \theta_2)\dot{\theta}_2 \end{pmatrix}, \end{aligned}$$

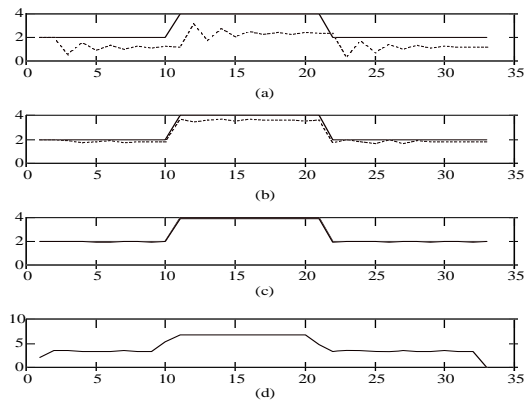


FIGURE 9.4. ILC algorithm behavior: (a) desired output and initial output; (b) desired output and output on the 5th trial; (c) desired output and output on 10th trial; (d) input signal on 10th trial.

$$C(x) = \begin{pmatrix} 13.1625 \sin \theta_1 + 4.3875 \sin(\theta_1 + \theta_2) \\ 4.3875 \sin(\theta_1 + \theta_2) \end{pmatrix}.$$

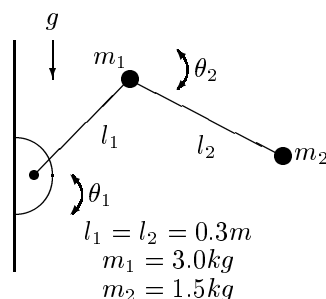


FIGURE 9.5. Two-joint manipulator.

To describe the learning controller for this system, first define the vectors

$$y_k = (x_k^T, \dot{x}_k^T, \ddot{x}_k^T)^T,$$

$$y_d = (x_d^T, \dot{x}_d^T, \ddot{x}_d^T)^T,$$

to represent the complete system output and desired trajectory, respectively at the k -th trial. The learning controller is then defined by

$$u_k = r_k - \alpha_k \Gamma y_k + C(x_d(0))$$

$$r_{k+1} = r_k + \alpha_k \Gamma e_k$$

$$\alpha_{k+1} = \alpha_k + \gamma \|e_k\|^m$$

Here Γ is a fixed feedback gain matrix that has been made time-varying through the multiplication by the gain α_k . The signal r_k can be described as a time-varying reference input. The adaptation of α_k combines with $r_k(t)$ to form the ILC part of the algorithm. Notice that with this algorithm we have combined conventional feedback with iterative learning control. Beyond these definitions, the operation of the system is the same as for any learning control system. A reference input r_0 and an initial gain α_0 are chosen to drive the manipulator during the first trial. At the end of each trial, the resulting output error e_k is used to compute the reference r_{k+1} and the gain α_{k+1} for the next trial. The basic idea of the algorithm is to make α_k larger at each trial by adding a positive number that is linked to the norm of the error so that when the algorithm begins to converge, the gain α_k will eventually stop growing. The convergence proof depends on a high-gain feedback result [160].

To simulate this ILC algorithm for the two-joint manipulator described above we used first-order highpass filters of the form

$$H(s) = \frac{10s}{s + 10}$$

to estimate the joint accelerations $\ddot{\theta}_1$ and $\ddot{\theta}_2$ from the joint velocities $\dot{\theta}_1$ and $\dot{\theta}_2$, respectively. The gain matrix $\Gamma = [P \ L \ K]$ used in the feedback and learning control law was defined by

$$P = \begin{pmatrix} 50.0 & 0 \\ 0 & 50.0 \end{pmatrix}, L = \begin{pmatrix} 65.0 & 0 \\ 0 & 65.0 \end{pmatrix}, K = \begin{pmatrix} 2.5 & 0 \\ 0 & 2.5 \end{pmatrix}.$$

These matrices satisfy a Hurwitz condition required for convergence. The adaptive gain adjustment in the learning controller uses $\gamma = .1$ and α_0 is initialized to 0.01.

Figure 9.6 shows the resulting trajectory of θ_1 for several trials. It also shows the desired trajectory. On the first trial the observed error is quite large, as expected, given that we have no knowledge of the system dynamics, other than the initial condition. However, only one iteration is needed to make the output distinctly triangular and by the eighth trial the system is tracking the desired response almost perfectly. Thus the technique has derived the correct input signal needed to force the system to follow a desired signal.

9.4 The Literature, Context, Terminology of ILC

9.4.1 Classifications of ILC Literature

The field of iterative learning control has a relatively small, but steadily growing literature. Table 9.1 gives a topical classification of general results in the field and Table 9.2 lists references that are specific to robotics and other applications. The ordering of the references in these two tables is roughly chronological and references may be listed more than once, depending on their coverage of topics. Also, the classification of a given paper into a particular category(ies) reflects this author's impression of the paper and is not necessarily the only possibility. Many of the references listed were obtained from the Engineering Index and the INSPEC electronic databases using a search strategy defined by: "control" AND "learning" AND "iterative." This is, of course, a very restrictive search strategy and it is quite likely that we have missed some papers. Unfortunately, the large number of conferences and journals available today make it impossible to be aware of every contribution on the subject of iterative learning control, which is discussed in a large number of fields, from robotics, to artificial intelligence, to classical control, to neural networks. Further, the phrase "iterative learning

TABLE 9.1. Iterative Learning Control Literature: General Results

Category	Papers
General/tutorial	[160, 169, 140, 162] [96, 177, 178, 23, 145]
Linear systems	
Adaptive/identification	[183, 185, 160, 191, 143] [162, 134, 72, 70]
Discrete-time	[222, 241, 202, 5, 10] [23, 190]
Direct learning	[112, 113, 239] [244, 115, 243, 246]
Frequency-domain	[87, 146, 108] [98, 153, 66, 1, 136] [103, 144, 68, 237]
General	[229, 222, 157, 158] [83, 26, 142]
Multivariable	[73, 184]
Non-minimum phase	[4, 196]
Norm-optimal	[8, 7, 10]
Periodic paramters	[174, 101]
Time-delay systems	[217, 90, 241, 91, 181]
Time-invariant	[15, 14, 12, 17, 21] [167, 166, 163, 152, 25] [150, 179, 182]
Time-varying	[15, 11, 89, 91, 121]
Two-dimensional systems	[75, 126, 139, 6, 178] [9, 3, 195]
Repetitive control	
General	[140, 145]
Linear	[156, 80, 81, 223, 211, 228] [128, 98, 97, 209, 3, 203]
Discrete	[223, 43, 27]
Nonlinear/stochastic	[151, 211, 122]
Convergence	
Linear	[21, 88, 99] [91, 210, 130]
High-gain/nonlinear	[111, 21, 24, 168] [176, 141, 149, 148]
Robustness	
Linear	[202, 210, 58, 61]
Nonlinear	[85, 86, 41, 247] [33, 218, 32, 31] [240, 46, 247] [200, 219]
Initial conditions	[129, 40, 39, 34]
No-reset	[208, 170]
Current cycle feedback	[242, 245, 177, 135, 41] [39, 58, 36, 170, 42]
Higer-order ILC	[21, 217, 64, 100, 36] [34, 100, 42]
Neural networks	[116, 165, 234, 230] [30, 224, 51]
Nonlinear	
Discrete	[101, 199, 141, 44, 45]
General	[84, 215, 154, 161, 141] [201, 104, 135, 48, 106, 238, 28] [76, 180, 49, 123, 52]
Inversion	[235, 221, 236, 187]

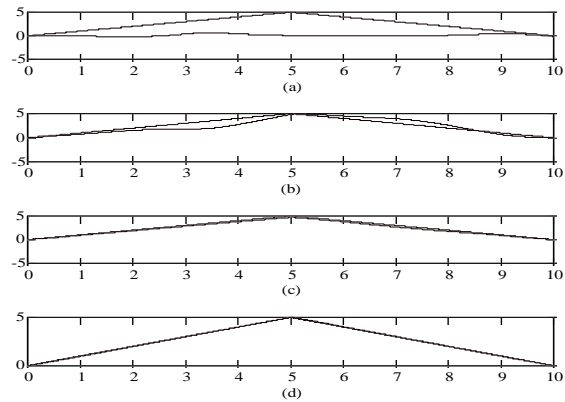


FIGURE 9.6. System response for joint angle θ_1 : (a) desired output and initial output; (b) desired output and output after 2nd trial; (c) desired output and output after 4th trial; (d) desired output and output after 8th trial.

control” has only recently become the standard phrase used to describe the ILC approach and many of the early papers dealing with ILC do not have all three of these terms in their title. It is inevitable that some work will be inadvertently omitted. Thus, the citation list has been limited to those works with which this author is familiar. Note also that Tables 9.1 and 9.2 list only those papers dealing with the ILC approach as we have defined it. As we discuss below, learning methods in control cover a broader spectrum than ILC. References dealing with other learning methods and concepts are not included.

At the risk of offending those who are left out and at the risk of appearing to lose impartiality, it is also possible to discuss the literature by author. Such a classification is useful, however, as it gives an idea of the level of interest in the field. The concept of iterative learning control in the sense of Figure 9.1 was apparently first introduced by Uchiyama in 1978 [229]. Because this was a Japanese language publication it was not widely known

TABLE 9.2. Iterative Learning Control Literature: Robotics and Applications

Category	Papers
Robotics	
General	[15, 13, 16, 55, 235, 11] [109, 222, 157, 158, 19, 54] [111, 112, 77, 82, 113, 164] [56, 24, 174, 155, 168, 205] [22, 95, 248, 249, 204, 146] [96, 67, 102, 242, 207] [125, 117, 194, 124, 118, 18]
Elastic joints	[114, 59]
Flexible links	[212, 213, 148, 57, 147]
Cartesian coordinates	[225, 78]
Neural networks	[230, 30, 224]
Cooperating manipulators	[254, 216]
Hybrid force/position control	[110, 105, 149]
Nonholonomic	[175]
Applications	
Vehicles	[232, 231, 138]
Chemical processing	[251, 29, 50] [132, 37, 133]
Machining/manufacturing	[137, 119, 47, 250, 79, 120]
Mechanical systems	[228, 131, 187, 188] [198, 66, 65, 68] [179, 253, 159, 189]
Miscellaneous	[127, 186, 93, 74] [92, 35, 62, 38] [192, 63, 197]
Nuclear reactor	[107]
Robotics demonstrations	[110, 109, 19, 205, 248, 204] [146, 105, 96, 207, 59, 175] [78, 194, 30]

in the West until the idea was developed by the Japanese research group of Arimoto, Kawamura, and Miyazaki, particularly through the middle to late 1980's [15, 13, 16, 14, 110, 11, 12, 17, 109, 111, 114, 112, 113, 85], [86, 153, 18, 115]. This author was involved in several new ILC results in the late 1980's to early 1990's [164, 165, 160, 167, 166, 168] [163, 161, 234, 169, 170], including the book *Iterative Learning Control for Deterministic Systems* [162]. Also during this period, a research group at the Dipartimento di Infomatica, Sistemistica in Genoa, Italy, including Bondi, Lucibello, Ulivi, Oriolo, Panzieri and others, was quite active [24, 146, 59, 175, 147, 149, 150, 148]. Other early researchers included: Mita, et al. [156, 157, 158], Craig, et al. [55, 54, 56], and Hideg, et al. [87, 108, 88, 89, 90, 91]. Also of note is the work of Tomizuka, et al. in the area of repetitive control [223, 43, 228, 105, 98, 47, 97, 209]. Other active researchers in ILC include Horowitz, et al. [205, 154, 95, 96], Sadegh, et al. [205, 204, 203, 78], Saab [201, 199, 202, 200], and C.H. Choi, et al. [250, 104, 48, 49]. A group at KAIST, centered around Professor Bien, one of the pioneers in ILC research, has made numerous contributions [174, 21, 22, 101, 129, 130, 181]. Another leading research group is based around Professor Longman at

Columbia University [155, 184, 183, 185, 26, 141, 143, 134, 67, 102, 198], [68, 142, 99, 136, 237, 103, 144, 66, 65, 140, 145] and Professor Phan of Princeton [192, 190, 189, 187, 188, 186, 191, 70]. Recently the work of Amann, Owens, and Rogers has produced a number of new ideas [8, 9, 7, 177, 178, 5, 10, 4, 176]. Finally, we mention a very active research group in Singapore centered around Jian-Xin Xu and Yangquan Chen [242, 245, 241, 41, 40, 239, 39, 240], [83, 36, 37, 38, 35, 62, 63, 42, 138, 219, 244, 243, 246], [33, 34, 218, 32, 31]. It is interesting to note that at the 2nd Asian Control Conference, held in Seoul in July 1997, over thirty papers were devoted to iterative learning control. This was nearly five percent of all the papers presented. Thus we see that ILC has grown from a single idea to a very active area of research.

9.4.2 Connections to Other Control Paradigms

Before proceeding to discuss ILC algorithms that have been proposed it is useful to consider the difference between iterative learning control and some of the other common control paradigms. We will also clarify some of the terminology related to ILC.

Consider Figure 9.7, which shows the configuration of a unity feedback control system, possibly multi-input, multi-output. Here P represents the system to be controlled and C denotes the controller. The control system

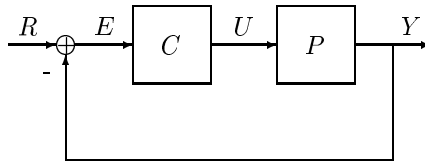


FIGURE 9.7. Unity feedback control system.

operates by measuring Y , the output of the plant, and comparing it to the reference input R . The error E is then input to the controller, which computes an appropriate actuating signal U . This signal is then used to drive the plant. A typical design problem is to specify a controller for a given plant, so that the closed-loop system is stable and the output tracks a desired trajectory with a prescribed set of transient performance characteristics (such as overshoot, settling-time, rise-time, steady-state error, etc.). Using this figure we may describe a number of controller design problems:

1. *Feedback control via pole placement or frequency domain techniques:* These problems can be described as:

Given P and R , find C so that the closed-loop has a prescribed set of poles, frequency characteristics, or steady-state error properties.

This problem is obviously different from the ILC approach, which, as we have noted, is not a feedback methodology. The iterative learning controller cannot affect the system poles.

2. *Optimal control*: Most optimal control problems can be generally described as:

$$\min_C \|E\|$$

subject to: P and R given and to constraints on U .

In optimal control we conduct an *a priori* design, based on a model of the system. If the plant changes relative to the model then the optimal controller will cease to be optimal. Further, the optimal controller operates in a feedback loop. Note, however, that in the case of a stable plant it may be possible to design an ILC system that produces the same output as an optimal controller, because both methods are concerned with the minimization of a measure of the error. The difference is that the ILC algorithm achieves this by injecting the optimal input U^* into the system, as opposed to forming it by processing the error in real-time. This can be illuminated by referring to Figure 9.8, which emphasizes in a different way how the ILC scheme can be viewed as an open-loop control strategy. The key point is that ILC is a way to derive the signal U^* , using information about past behavior of the system. Figure 9.8 points out the obvious absence of an explicit controller in the ILC approach.

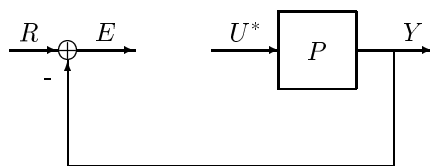


FIGURE 9.8. Another representation of the ILC approach.

3. *Adaptive control*: On the surface one might think that ILC and adaptive control were very similar. The key difference however, is exactly the difference between Figure 9.7 and Figure 9.8 – the fact that one lacks an explicit controller. Iterative learning control is different from conventional adaptive control in that most adaptive control schemes

are on-line algorithms that adjust the controller's parameters until a steady-state equilibrium is reached. This allows such algorithms to essentially implement some type of standard feedback controller such as in item (1) above while dealing with parametric uncertainty or time-varying parameters in the plant. Of course, it is true that if the plant changes in a learning control scheme, the learning controller will adapt by adjusting the input for the next trial, based on the measured performance error of the current trial. However, in a learning control scheme, it is the commanded reference input that is varied (in an off-line fashion), at the end of each trial or repetition of the system, as opposed to the parameters of a controller.

4. *Robust Control*: Referring again to Figure 9.7, robust control is a set of design tools to deal with uncertainty in the plant. With this broad definition one could call ILC a robust control scheme, because ideally the ILC algorithm does not require information about the plant. Again, however, ILC is not the same as robust control in that there is no explicit controller.
5. *Intelligent Control*: Recently a number of control paradigms have developed that can be loosely gathered under the umbrella of so-called intelligent control techniques. These include artificial neural networks, fuzzy logic, and expert systems. One thing all these have in common is that they usually involve learning in some form or another. As such, the phrase "learning control" often arises and in general ILC as we have described it here can also be classified as a form of intelligent control. However, it should be made clear that ILC is a very specific type of intelligent control and involves a fairly standard system-theoretic approach to algorithms, as opposed to the artificial intelligence- or computer science-oriented approaches often found in neural nets, fuzzy logic, and expert system techniques.

Although recently the word "learning" has become almost ubiquitous, it is, unfortunately, often the cause of misunderstanding. In a general sense, learning refers to the action of a system to adapt and change its behavior based on input/output observations. We have noted in [162] that in the cybernetics literature the term learning has been used to describe the ability of a system to respond to changes in its environment. Many systems have this ability at different levels. Thus, when using the term learning control, we must define the meaning carefully. For ILC we use the word learning in the sense of the architecture shown in Figure 9.1, where the concern is with iteratively generating a sequence of input trajectories, $u_{k+1} \rightarrow u^*(t)$. In addition to the word "learning," further confusion can arise when placing it together with the word "control." This is because the term "learning control" is not unique in the control literature. Researchers in the fields of adaptive control [226], stochastic control and cybernetics [206, 172, 173],

and optimal control [193] have all used the term learning control to describe their work. Most of these references, however, refer to learning in the sense of

adapting or changing controller parameters on-line, as opposed to the off-line learning of ILC. Other authors have considered the general problems of learning from a broader view than either ILC or adaptation of parameters. [71, 227] are early works on the subject. [233] gives a recent mathematical perspective. Several researchers have also considered learning control as a special case of learning in general, in the context of intelligent systems [94, 214, 53, 252]. A particularly visionary work in this regard is [2]. We mention one other area in which the phrase “learning control” arises: reinforcement learning control. Reinforcement learning controllers are sophisticated stochastic search engines and are very much an ILC technique in the sense that we have defined ILC. They work by evaluating the outcome of an action after the action and its effect are over. The next action is then chosen based on the outcome of the previous action. Because of the stochastic nature of this approach we will not discuss it further, but refer the reader to [20, 69, 220, 161]. Likewise we will not consider any explicitly stochastic learning controllers [206, 60].

Finally, after saying what ILC is not, we should say what ILC is. Terms used to describe the process of Figure 9.1 include “betterment process” (Arimoto’s original description), “iterative control,” “repetitive control,” “training,” and “iterative learning control.” As in the discussion about the term “learning,” one must be careful to define what is meant in any particular usage of a word or phrase. For instance, the term “repetitive control” is used to mean ILC but is also used to describe the control of periodic systems (see the discussion in the next section). Also, in [248] the term “virtual reference” is introduced to describe the optimal input signal derived by the learning controller. This emphasizes the fact that ILC algorithms produce the output of an optimal prefilter and what we are doing in essence is to compute a “virtual” or “not-real” input to try to fool the system into going to where we want it to go.

9.5 ILC Algorithms and Results

In this section we will describe some of the work in the field of iterative learning control. Our comments will roughly follow the organization of Tables 9.1 and 9.2, although some topics will be discussed in a different order. Also, due to space limitations we cannot individually address each paper listed in the references and, as in the discussion of the table, the same caveat applies: it is inevitable that some work will be inadvertently omitted. However, the results we describe will give a reasonably complete picture of the status of the field of iterative learning control. We also refer

the reader to [162], [96], [177], [23], and [145], each of which contains a significant amount of tutorial information.

9.5.1 Basic Ideas

The first learning control scheme proposed by Arimoto, et al. involved the derivative of the error $e_k(t) = y_d(t) - y_k(t)$ [15]. Specifically, the algorithm had the form

$$u_{k+1} = u_k + \Gamma \dot{e}_k.$$

This is the continuous-time version of the algorithm we gave in the linear example above:

$$u_{k+1}(t) = u_k(t) + \Gamma e_k(t+1).$$

For the case a linear time-invariant (LTI) system, with signals defined over the interval $[0, t_f]$, and with a state-space description (A,B,C), Arimoto, et al. showed that if $CB > 0$ and if the induced operator norm $\|I - CB\Gamma\|_i$ satisfies

$$\|I - CB\Gamma\|_i < 1,$$

and some initial condition requirements are met, then

$$\lim_{k \rightarrow \infty} y_k(t) \rightarrow y_d(t),$$

in the sense of the λ -norm, defined as

$$\|x(t)\|_\lambda = \sup_{0 \leq t \leq t_f} \{e^{-\lambda t} \max_{1 \leq i \leq r} |x_i|\},$$

where $x(t)$ is an r -dimensional vector. Arimoto, et al. also gave convergence conditions for this type of learning control algorithm when it was applied to time-varying plants and certain nonlinear models encountered in robotics. In subsequent papers Arimoto and his co-workers proposed a variety of different learning control algorithms. The primary differences between the various approaches they have developed is in how the error is utilized in the learning control algorithm. The most general linear algorithm presented is found in [11], where the input is updated according to

$$u_{k+1} = u_k + \Phi e_k + \Gamma \dot{e}_k + \Psi \int e_k dt.$$

This algorithm essentially forms a PID-like system for processing the error from the previous cycle, while maintaining a linear effect on the past input signal.

Much of the early work in ILC focused on linear operators, with learning laws of the form:

$$u_{k+1} = T_u u_k + T_e (y_d - y_k),$$

where T_u and T_e are both linear operators. This is really the most general linear algorithm we might consider because it allows separate weighting of both the current input and the current error. In [162] we proved the following sufficient condition:

Theorem For the LTI plant $y_k = T_s u_k$, the proposed LTI learning control algorithm

$$u_{k+1} = T_u u_k + T_e (y_d - y_k)$$

converges to a fixed point $u^*(t)$ if

$$\|T_u - T_e T_s\|_i < 1.$$

The fixed point u^* is given by

$$u^*(t) = (I - T_u + T_e T_s)^{-1} T_e y_d(t)$$

and the resulting fixed point of the error is given by

$$e^*(t) = \lim_{k \rightarrow \infty} (y_k - y_d) = (I - T_s (I - T_u + T_e T_s)^{-1} T_e) y_d(t),$$

for $t \in [t_0, t_f]$.

The gist of almost all of the ILC results is related to proper selection of the learning operators T_u and T_e for specific classes of systems. In the remainder of this section we will discuss some of these results.

Optimization-Based Approaches

One approach taken by some authors is to pick T_e so as to force the convergence to follow some gradient of the error. For example, in [222] the discrete-time learning control algorithm

$$u_{k+1}(t) = u_k(t) + G e_k(t+1),$$

is used, with the gain G optimized using gradient methods to minimize the quadratic cost of the error

$$J = \frac{1}{2} e_k^T(i+1) Q e_k(i+1).$$

between successive trials. The authors consider several techniques for choosing G , specifically using the steepest-descent, Gauss-Newton, and Newton-Raphson methods. The first two result in a constant gain G , giving a learning controller with exactly the same form as Arimoto, et al. For the Newton-Raphson method the result is a time-varying gain G_k which is different for each trial. In [73] the update algorithm has the form

$$u_{k+1} = u_k + \epsilon_k T_p^* e_k,$$

where T_p^* is the adjoint operator of the system and ϵ_k is a time-varying gain computed from the error and the adjoint to provide a steepest-descent minimization of the error at each step of the iteration. This work also explicitly considered multivariable systems.

Norm-Optimal ILC

A more recent approach to ILC algorithm design has been developed by Amann, et al. [8, 7, 5, 10]. They compute the change in the input so as to minimize the cost

$$J_{k+1} = \|e_{k+1}\|^2 + \lambda \|u_{k+1} - u_k\|^2.$$

It is shown that the optimal solution produces a similar result to that of [73]:

$$u_{k+1} = u_k + G^* e_{k+1},$$

where G^* is the adjoint of the system. Note however, that in Amann's algorithm the error is from the current cycle (e.g., e_{k+1}). Thus the resulting ILC algorithm can be thought of as combining previous cycle feedback (a feedforward effect) with current cycle feedback (a feedback effect). We will discuss this more below. A discussion of the convergence of norm-optimal ILC for discrete-time systems is given in [5, 10].

Frequency-Domain Approaches

Several authors have considered ILC from the frequency domain perspective. In perhaps the first of these, [157, 158], the input update law is defined by

$$U_{k+1}(s) = L(s)[U_k(s) + aE_k(s)].$$

Convergence is shown in the sense of the L_2 norm (time or frequency), although in [162] it was noted that this algorithm will produce a non-zero error. Arimoto's original algorithm is considered in the frequency domain in [1]. In [146] ILC algorithms with completely independent operators T_e and T_u are designed in the frequency domain. Hideg, et al. have considered a number of frequency domain results for ILC [87, 108]. An interesting approach to ILC based on the discrete Fourier transform is given in [153]. The DFT is used to get a locally linear representation of the system, which is then inverted. The results becomes equivalent to those of this author [162] described in the next paragraph. By far, the definitive works on the use of frequency domain techniques in ILC have been made by Longman, et al. See [68, 237, 103, 144], for example. A key concept introduced in these works is the idea of phase cancellation through learning. An excellent review of this approach can be found in [145].

Discrete-Time Systems

A number of researchers have considered ILC specifically for discrete-time systems, primarily motivated by the fact that all practical implementations will result in a discrete-time ILC algorithm. We have already mentioned the early work of [222]. More recent analysis of the convergence properties of the Arimoto-type "D-algorithms" was given in [202]. The issue of instability

in ILC implementations due to the sampling delay has been considered in [241, 23].

It is interesting to consider the discrete-time linear case [162]. Consider again Figure 9.1 and define

$$\begin{aligned} u_k &= (u_k(0), u_k(1), \dots, u_k(N-1)), \\ y_k &= (y_k(m), y_k(m+1), \dots, y_k(N-1+m)), \\ y_d &= (y_d(m), y_d(m+1), \dots, y_d(N-1+m)), \end{aligned}$$

where k denotes the trial, m is the relative gain of the linear plant, and N is the length of the trial. We will suppose that $m = 1$. Also, we will use the truncated l_∞ -norm, given by

$$\|x\|_\infty = \max_{1 \leq i \leq N} |x_i|$$

and the corresponding induced norm for a matrix H is given by

$$\|H\|_i = \|H\|_\infty = \max_i \left(\sum_{j=1}^N |h_{ij}| \right).$$

The linear plant can then be described by $y_k = H u_k$, where H is a matrix of rank N whose elements are the Markov parameters of the plant:

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

For this situation, consider the ILC algorithm:

$$u_{k+1} = u_k + A e_k,$$

where

$$A = \begin{bmatrix} \alpha_1 y_d(1) & 0 & \dots & 0 \\ \alpha_2 y_d(2) & \alpha_1 y_d(1) & \dots & 0 \\ \alpha_3 y_d(3) & \alpha_2 y_d(2) & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ \alpha_N y_d(N) & \alpha_{N-1} y_d(N-1) & \dots & \alpha_1 y_d(1) \end{bmatrix}.$$

In [162] it was shown that there exist gains α_i such that $\|e_k\|_\infty \rightarrow 0$. Further, it is possible to rearrange this algorithm

$$u_{k+1} = u_k + A e_k$$

into the form

$$u_k = A_k y_d,$$

where A_k is interpreted as an approximate inverse matrix that is updated at the end of each trial according to

$$A_{k+1} = A_k + \Delta A_k,$$

with

$$\Delta A = \begin{bmatrix} \alpha_1 e_k(1) & 0 & 0 & \dots & 0 \\ \alpha_1 e_k(2) & \alpha_2 e_k(1) & 0 & \dots & 0 \\ \alpha_1 e_k(3) & \alpha_2 e_k(2) & \alpha_3 e_k(1) & \dots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ \alpha_1 e_k(N) & \alpha_2 e_k(N-1) & \alpha_3 e_k(N-2) & \dots & \alpha_N e_k(1) \end{bmatrix}.$$

The same result in [162] that shows convergence of the error also shows that $A_k \rightarrow H^{-1}$.

The preceding discussion highlights a key result in ILC: the essential nature of ILC algorithms is to derive the output of the best possible inverse of the system dynamics. The same result has appeared in a variety of forms in several papers. In the case of linear systems, related continuous-time results are found in [160, 169, 157, 156].

Various Classes of Systems

Much of the work in ILC has focused on applying the basic ILC algorithm to different classes of systems. We have discussed discrete-time linear systems above. Other early results in ILC for *linear systems* were given by Arimoto, et al. in various papers [15, 14, 11, 12, 17]. [19] considers learning control schemes similar to Arimoto et al., with application to linear robotic manipulator models. [152] uses a coprime factorization approach to ILC design for linear systems. [150] formulates an ILC algorithm that involves learning the initial conditions of the plant. A novel application of ILC for linear systems is given in [25], which formulates a generalized predictive controller such as often used in process control. A particularly thorough analysis is given by Amann, et al. in [9], which gives an H_∞ approach to the design of ILC algorithms. Many of the papers by Arimoto, et al. also considered *time-varying systems*, as did the results in [174, 101]. [174] considers a class of linear time-varying plants. They use a parameter estimator together with an inverse system model to generate the new input for each trial. Their technique is also applied to the learning control of a robotic manipulator which looked at periodic variation of plant parameters. [89] considered general LTV multivariable systems. We also mention specifically [162], which explored the connections between *adaptive control* and ILC for linear systems. Other investigations along these lines include [183, 185, 191, 143, 134, 72, 70] *Non-minimum phase* systems have been discussed in [162, 4, 196]. The work in [196] uses the Nevanlinna algorithm of H_∞ control to design an ILC algorithm of the form

$$U_{k+1}(s) = P(s)U_k(s) + Q(s)E_k(s)$$

in the frequency domain. This algorithm is shown to converge, but, as shown in the more general work of [4], the presence of the RHP zero in the plant causes slow convergence and a non-zero error. Another class of problems that has been considered for ILC is systems that exhibit *time-delay* [217, 90, 241, 91, 181]. The approach in [90] is to include a delayed version of the previous cycle error in the update law, resulting in an ILC algorithm of the form

$$u_{k+1}(t) = u_k(t) + \alpha_1 \dot{e}_k(t) + \alpha_2 e_k(t) + \alpha_3 e_k(t - \tau),$$

where τ is the time delay of the linear system. For this system the paper demonstrates conditions for convergence. Another important issue is the ability of an ILC algorithm to converge in the face of *actuator saturation* constraints. Two works that have considered this problem are [141, 49].

Two-Dimensional Systems Theory

Note that when we combine the plant, expressed in dynamical form as $y_k(t+1) = T_p u_k(t)$, with the learning controller update law $u_{k+1}(t) = T_u u_k(t) + T_e [y_d(t+1) - y_k(t+1)]$ and if we change the notation slightly we end up with the equation:

$$y(k+1, t+1) = T_p (T_u u(k, t) + T_e [y_d(t+1) - y(k, t+1)])$$

This is a linear equation indexed by two variables: k and t . As such it is possible to apply two-dimensional systems theory to analyze the convergence and stability properties of the ILC system. This has been investigated by several researchers. For example, in [126] the problem is considered for a standard (A, B, C) state-space representation of linear systems. If $T_u = I$ and $T_e = K$, a constant gain matrix, it is shown that the ILC algorithm converges if $I - CBK$ is stable. A particularly in-depth discussion of this approach is found in [75]. [139] considers robustness of ILC algorithms from the perspective of two-dimensional system theory. Amann, et al. have also considered two-dimensional analysis [6, 178, 9], using results based on Lyapunov stability analysis techniques [195]. [3] present an analysis of repetitive control (see below) using two-dimensional system theory.

Convergence and Robustness

Much of the past and recent work in ILC has focused on demonstrating convergence of ILC algorithms and analyzing the robustness of the convergence properties. Of course, convergence is implicitly considered in every paper on ILC appearing in the references. However, some authors have explicitly considered these issues. In [99] the phenomena of initial convergence followed by divergence is explored and techniques are given to avoid the problem. [88] describes the fact that the actual norm used can distort the true picture of the stability of the ILC algorithm. [91] considers convergence

for systems with time delay properties. In a result that parallels a number of results about the performance of ILC, [210] proves that there exist no bounded operator T_e such that the update law $u_{k+1} = u_k + T_e e_k$ results in exponential convergence for linear systems. However, the paper shows that by introducing a digital controller it is possible to achieve exponential convergence in exchange for certain non-zero residuals. Much of the most well-developed work regarding convergence is by Amann, et al. Be careful to note, however, that the ILC update law used in much of this work is not the same as the one we used in most of this paper. Rather, Amann's work uses current cycle feedback along with previous cycle feedback. One of the main approaches to studying convergence in ILC is to invoke high-gain results. In the robotics example given above, the convergence proof is based on the fact that when the adaptive gain α_k gets big enough then the ILC algorithm will converge. Amann explores this carefully in several papers (see [8], for example). Other references in which a high-gain condition is invoked are listed in Table 9.1. One of the new directions for the study of convergence is to consider modification to the ILC update law. For instance, in [21] the authors use errors from more than one previous cycle to update the input. By doing this they show that convergence can be improved. This is called higher-order ILC.

Associated with the question of convergence is that of robustness. Generally, the results that are available consider a convergent ILC algorithm and study how robust its convergence properties are to various types of disturbances or uncertainties. Much recent work has focused on mis-match in initial condition from trial to trial. [86] computed error bounds due to the effects of state and output noise as well as initial condition mis-match. In [129] a very good analysis of this effect is given and it is shown how undesirable effects due to mismatch can be overcome by utilizing a previous cycle pure error term in the learning controller. [202] considers robustness and convergence for a "D-type" ILC algorithm. A condition is given for a class of linear systems to ensure global robustness to state disturbances, measurement noise, and initialization errors. Another approach to robustness has been to combine current cycle feedback with previous cycle feedback [41, 40]. In particular, in [40] a typical "D-type" ILC algorithm is combined with a scheme for learning the initial condition. This information is then used to offset the effect of the initial condition mis-match. For linear systems, [139] shows how to use H_∞ theory in a two-dimensional systems theory framework to address modelling uncertainty.

9.5.2 Nonlinear Systems

Most of the basic results described above were derived primarily for linear systems. However, researchers have also considered the learning control problem for different classes of nonlinear systems. Of particular interest to many researchers are the classes of nonlinear systems representative of

robotic manipulator models. However, ILC for nonlinear systems has also been considered independent of robotics.

Suppose now that our system is defined by

$$y_k(t) = f_P(u_k(t), t),$$

where f_P is a nonlinear function. According to our problem statement we seek a system f_L such that the sequence

$$u_{k+1}(t) = f_L(u_k(t), y_k(t), y_d(t), t) \rightarrow u^*(t),$$

where the optimal input $u^*(t)$ minimizes the norm of the final error

$$\|y_d(t) - f_P(u^*(t), t)\|.$$

To deal with this problem, we can seek a contraction requirement to develop sufficient conditions for convergence. This is the approach of almost every result available in the literature that deals with nonlinear learning control. Suppose that we have $u_k(t) \in U$, where U is an appropriately defined space. Then for the learning algorithm

$$u_{k+1} = f_L(u_k, f_P(u_k), y_d),$$

we will obtain convergence if for all $x, y \in U$ there exists a constant $0 < \rho < 1$ so that

$$\|f_L(x, f_P(x), y_d) - f_L(y, f_P(y), y_d)\| \leq \rho \|x - y\|.$$

The question that can then be posed is the following: what conditions on the plant and the learning controller will ensure that the iteration is a contraction mapping? The papers listed in Table 9.1 present various answers to this question and are primarily distinguished by the different restrictions that are placed on the system. Several representative examples are given in the next few paragraphs.

[235, 221, 236] apply the idea of learning control to the problem of determining the inverse dynamics of an unknown nonlinear system. In [221] the following ILC algorithm is proposed:

$$u_{k+1} = u_k + (y_d - y_k).$$

This is a simple linear learning control algorithm, with unity weighting on both the current input and the current error. If $y_k(t) = f_P(u_k)$, where $f_P(\cdot)$ is continuous on the interval of interest, then convergence is guaranteed if

$$\|I - f'_P(u)\| < 1$$

for all inputs $u \in S$, where S is convex subset of the space of continuous function and $f'_P(u)$ is the derivative of f_P with respect to its argument

u. This result gives a class of systems for which the ILC algorithm will converge.

However, the previous result is very general and is not always easily checked. Less general results can be obtained by restricting the plant and the ILC algorithms to have a more specific structure. Let the plant be described by

$$\begin{aligned}\dot{x}_k &= a(x_k, t) + b_p(t)u_k, \\ y_k &= c(x_k, t) + d_p(t)u_k,\end{aligned}$$

with $a(x, t)$ and $c(x, t)$ Lipschitz in their arguments. Let the ILC update law be

$$\begin{aligned}\dot{v}_k &= A_c(t)v_k + B_c(t)e_k, \\ u_{k+1} &= C_c(t)x_k + D_c(t)e_k + u_k.\end{aligned}$$

For this setup, convergence can be shown if [215]

$$\|I - d_p(t)D_c(t)\| < 1.$$

Notice that this result depends only on the direct transmission terms from the plant and the learning controller.

A similar results was given in [84]. For the system

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

with a learning algorithm given by

$$u_{k+1} = u_k + L(y_k)(\dot{y}_d - \dot{y}_k),$$

it is shown that convergence is obtained if $f(\cdot)$ and $B(\cdot)$ are Lipschitz and $L(\cdot)$ satisfies

$$\|I - L(g(x, t), t)g(x, t)B(x, t)\| < 1.$$

We can see that this convergence condition has the same form as in the other examples.

An alternative to applying contraction mapping conditions is to use Lyapunov analysis. This is typical in the analysis of learning in neural networks, which can be considered a type of learning control problem. This is also the approach taken in the convergence analysis of a novel learning control scheme proposed in [154]. The technique is based upon a new method of nonlinear function identification. As in the case of contraction mapping techniques, however, the use of Lyapunov analysis techniques requires that we place varying levels of assumptions on the plant and the learning controller in order to obtain useful convergence conditions. The three examples

above give an idea of the types of analysis and results available for nonlinear ILC. Interested readers can consult the references listed in Table 9.1 for additional information. We have previously noted in [162] that there is not a unifying theory of iterative learning control for nonlinear systems. This does not seem to have changed as of this writing. Results from contraction mapping or Lyapunov techniques usually provide sufficient, but general conditions for convergence, which must be applied on a case-by-case basis. To obtain more useful results it is often necessary to restrict our attention to more specific systems, as in the examples above. One example of this is the problem of learning control for robotic manipulators. By assuming that the nonlinear system has the standard functional form typical of a manipulator, researchers have been able to establish specific learning controllers that will converge. We mention as a final comment that many researchers have begun to use neural networks as one tool for nonlinear ILC. In most papers, for instance [234], by this author, the results show an approach that works, but there is little analysis to say why it works. Consequently, it is our view that the more general problem of ILC for nonlinear systems is still an open area of research.

9.5.3 Robotics and Other Applications

We have noted several times that robotics is the natural application area for ILC. Arimoto, et al.'s original work included a discussion of learning control for robotics and others have independently proposed similar learning and adaptive, motivated by robotic control problems [55, 56, 54]. It is beyond the scope of this single paper to discuss ILC for robotics in any suitable detail. The interested reader can refer to the references in Table 9.2 to identify papers dealing with different aspects of ILC and robotics. In particular, we recommend [96], which contains a good summary of ILC for robotics. Also, to get a flavor for the types of algorithms that are used, the reader may refer back to the representative example of an ILC algorithm applied to a simulated two-joint manipulator that was given earlier in the paper. See also [24], which first gave the high-gain feedback, model-reference approach to learning control that motivated the adaptive result presented in the example. It should be clear from Table 9.2 that researchers have considered a wide-variety of problems in robotics. It is also interesting to point out the large number of actual demonstrations of ILC using robots. Indeed, in [248] an incredible demonstration of an ILC algorithms is reported in which a robot arm iteratively learns to catch a ball in a cup (the Japanese Kendama game).

In addition to robotics, iterative learning control has been applied to an increasing number of applications, as shown in Table 9.2. Most of these have been to problems that involve repetitive or iterative operations. For instance, in chemical processing, ILC has been applied to batch reactor control problems, which is inherently an iterative activity [251, 50, 132].

Other applications emphasize learning on-line based on rejection of periodic or repeatable errors that occur during normal step changes or other operational procedures [232, 231, 127, 107]. Many of these consider periodic systems and apply ILC to solve disturbance rejection or tracking problems, including applications for controlling peristaltic pump used in dialysis [93], coil-to-coil control in a rolling mill [74], vibration suppression [92], and a nonlinear chemical process [29]. Motion control of non-robotic systems has also attracted a significant amount of research effort for ILC applications, including a scanner driver for a plain paper copier [253], a servo system for a VCR [131], a hydraulic servo for a machining system [47], an inverted pendulum [179], a cylindrical cutting tool [79], CNC machine tools [228, 119, 250, 120], and an optical disk drive [159]. [137] also describes an application to self-tuning a piezo-actuator. Following the next section we apply some new approaches to learning control to a gas-metal arc welding problem [170].

9.5.4 Some New Approaches to ILC Algorithms

In this section we will discuss three specific emerging areas in ILC research. The first is the intersection between repetitive control and ILC, which has led to the idea of “no-reset” or “continuous” ILC. The second area is the development of ILC algorithms that do not fit the standard form of $u_{k+1} = T_a u_k + T_e e_k$. We end with what has been called “direct learning control.”

Repetitive Control and No-Reset/Continuous ILC

Strictly speaking, repetitive control is concerned with cancelling an unknown periodic disturbance or tracking an unknown periodic reference signal [96]. The solutions that have been developed in the literature tend to focus on the internal model principle to produce a periodic controller [80]. As such, the repetitive controller is a feedback controller as opposed to the ILC scheme, which ultimately acts as a feedforward controller. Other differences include the fact that the ILC algorithms act on a finite horizon whereas the repetitive controllers are continuous and the fact that the typical assumption of ILC is that each trial starts over at the same initial condition whereas the repetitive control system does not start with such an assumption. Despite these differences however, it is instructive to consider the repetitive control strategy (a good summary of repetitive control design is found in [128]), because the intersection between the strict definitions of repetitive control and ILC is a fertile ground for research. It is also interesting to note that the topical breakdown between the two fields is very similar, with progress reported in the field of repetitive control for linear systems [156, 80, 81], multivariable systems [203], based on model-matching and 2-D systems theory [3], for discrete-time systems explicitly [97, 223, 43, 27], using frequency domain results [98, 211], nonlinear sys-

tems [151], and stochastic systems [43, 122]. Also note that the connections between ILC and repetitive control are not new. In particular, see many of the works by Longman, et al. ([145], for example), which make it clear that ILC and repetitive control are really the same beast.

To illustrate how ILC and repetitive control are similar, we consider a simulated vibration suppression problem. Consider a simple system defined by

$$y(t) = K_p(u(t) + T_d d(t))$$

where we define

$$T_d = G_d(s) = \frac{K_d e^{-T_d s}}{\tau_d s + 1}.$$

Thus the system consists of a constant gain plant with an additive disturbance that is delayed and attenuated before it is applied to the plant. We suppose $d(t)$ is periodic (sinusoidal) with a known frequency. Suppose we wish to drive the output of the system to zero. It is clear that after all transients have died away the output will be periodic with the same period as the disturbance. This suggests a way to apply an ILC-like algorithm to the problem. We simply view each cycle of the output as a trial and apply the standard ILC algorithm. Because the output is periodic we can expect to have no problem with initial conditions at the beginning of each “trial.” The typical ILC update would be:

$$u_{k+1}(t) = u_k(t) + T_e e_k(t).$$

However, assuming a period of N , we know that $u_{k+1}(t) = u_k(t + N)$ (after the transients have died out). Thus the update law is effectively:

$$u(t + N) = u(t) + T_e e(t)$$

or, alternately,

$$u(t) = u(t - N) + T_e e(t - N).$$

But, this is simply a delay factor, as depicted in Figure 9.9, and this last equation is what has been called “... the familiar repetitive control law...” in [203]. Thus our ILC algorithm reduces in this case to a repetitive controller. Regardless of the implementation of the algorithm, the resulting controller easily suppresses the disturbance from the output, as shown in Figure 9.10.

From this example one can see that ILC applied to a periodic, continuous-time situation is equivalent to repetitive control. This has been addressed recently by several researchers. The idea is called “no-reset” ILC in [208] because the system never actually starts, stops, resets, and then repeats. Rather, the operation is continuous. This suggests calling an ILC approach to such a control system to be a “continuous” iterative learning controller. Below we give an example of an extended version of the idea for a system

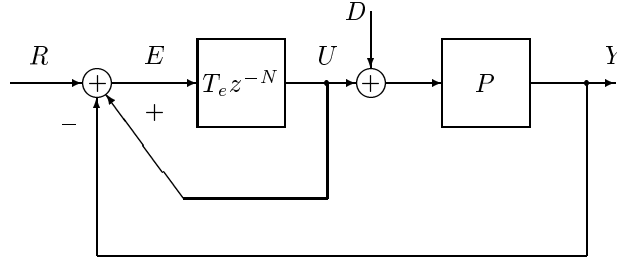


FIGURE 9.9. A periodic representation of the ILC algorithm.

that is not periodic, but rather is chaotic. As we comment in the conclusions, the study of continuous ILC algorithms is a very promising research area.

Current Cycle Feedback and Higher-Order ILC

The standard ILC algorithm that we have described generally has the form:

$$u_{k+1}(t) = T_u u_k(t') + T_e e_k(t'),$$

where $t' \in [t, t_f]$ and where we will assume for now that T_u and T_e are general operators and could possibly be nonlinear. There have been two distinct modifications to this equations that have been presented in the literature:

1. *Higher-Order ILC*: This was first suggested in [21] and has also been called “multi-period” ILC in [242, 245]. The algorithm has the form:

$$u_{k+1}(t) = T_u u_k(t') + T_{e_1} e_k(t') + T_{e_2} e_{k-1}(t') + \dots$$

That is, our update law is now looking back past the most recent previous cycle to bring in more data about the past performance of the system. In [21] it is noted that this modification can improve the rate of convergence. Note that this is also a natural algorithm when viewing the ILC process from the perspective of two-dimensional systems theory. Other works that consider higher-order ILC include [217, 64, 100, 36, 34, 100, 42].

2. *Current Cycle Feedback*: An idea that goes back at least as far as 1987 [83] is to combine iterative learning control with conventional feedback control. This has been done in a couple of different ways. Algorithms in [242, 245, 135, 41, 39, 58] and others update the input according to

$$u_{k+1}(t) = T_u u_k(t') + T_{e_{ff}} e_k(t') + T_{e_{fb}} e_{k+1}(t).$$

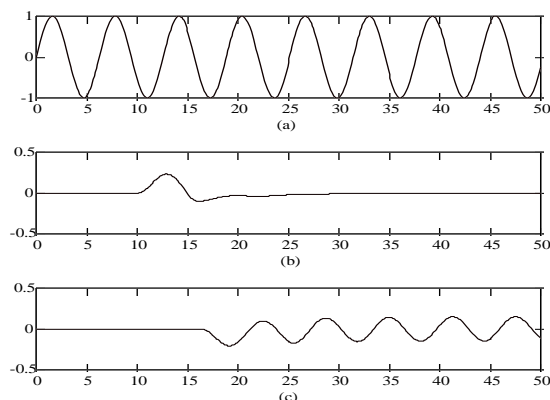


FIGURE 9.10. System response for continuous ILC example: (a) disturbance input; (b) output signal; (c) ILC produced input signal.

Thus we have simply added the error from the current cycle to the ILC algorithm. In such a scheme it is easy to show for a plant T_p and with $T_u = I$ the sufficient condition for convergence becomes:

$$\|(I + T_p T_{e_{fb}})^{-1} (I - T_p T_{e_{ff}})\| < 1.$$

Thus we see a combination of the normal ILC convergence condition with a more common feedback control type expression. Amann, et al. have given a slightly different form of the current error feedback algorithm:

$$u_{k+1}(t) = T_u u_k(t') + T_e e_{k+1}(t').$$

This expression does not explicitly use past cycle error information, but does keep track of the previous inputs. Most of the results on norm-optimal ILC by Amann, et al. use this form of an update algorithm. An obvious advantage of either of these current error feedback algorithms is that it is possible to ensure stability in both trial convergence and with respect to time relative to the closed-loop system. Others to consider current cycle feedback include [36, 170, 42].

Figure 9.11 gives a graphical representation of these two different strategies that shows pictorially where in time the values that are used in the update equation are taken from. Notice also that these two major modifications

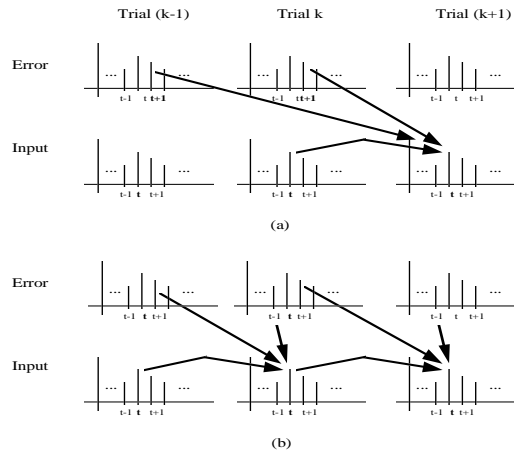


FIGURE 9.11. Two new ILC strategies: (a) higher-order learning control; (b) current cycle feedback.

can be combined to give what may be the most general form of an ILC algorithm:

$$u_{k+1}(t) = T_{fb}e_{k+1}(t) + T_u u_k(t') + T_{e_1}e_k(t') + T_{e_2}e_{k-1}(t') + \dots$$

This algorithm was considered by Xu in [36, 42]. With such an algorithm it may be possible to take exploit the advantages of both the current error feedback and the higher-order elements. This will certainly be an important area for additional research.

Direct Learning

One final category that we would like to address is what has recently been called “direct learning” [239]. The concern in this area of study is that

once an ILC algorithm has converged and is then presented with a new trajectory to follow it will lose any memory of the previous trajectory, so that if it is presented again it will have to re-learn. Of course, in practice we may keep track of the optimal input U^* corresponding to different desired outputs, but at a more fundamental level, this represents a shortcoming of the ILC approach. Arimoto, et al. considered this in some earlier papers [112, 113]. Also, in [162] an approach was given for “learning with memory.” Recently, however, the problem was posed in two interesting ways:

1. Can we generate the optimal inputs for signals that have the same shape and magnitude, but different time scales by learning only one signal [239, 243]?
2. Can we generate the optimal inputs for signals that have the same shape and time scales, but different magnitudes [246]?

At this time there are no definitive conclusions that can be drawn, but this will be an important research area in the future (see also [244, 115]).

9.6 Example: Combining Some New ILC Approaches

In the previous section we ended by describing a number of new approaches to ILC. In this section we give an example of an ILC algorithm that combines the ideas of current cycle feedback and continuous ILC to develop a controller for a nonlinear, chaotic system in which the goal is to force a prescribed periodic motion. We have noted that two of the fundamental assumptions associated with iterative learning control are that (1) each trial has the same length and (2) after each trial the system is reset to the same value. The learning controller we illustrate here does not require these assumptions. Our design is motivated by the problem of controlling a gas-metal arc welding process. In this process the time interval between detachments of mass droplets from the end of a consumable electrode is considered to be a trial. This interval, as well as the mass that detaches, is deterministically unpredictable for some operating points. Our control objective is to force the mass to detach at regular intervals with a uniform amount of mass in each detached droplet. Thus our problem can be cast in an iterative learning control framework where both the trial length and the initial conditions at the beginning of each trial are non-uniform. However, by careful consideration of when the trial ends, relative to when we desire the trial to end, it is possible to force the time of the trial to the desired value, using a cascaded, two-level iterative learning controller combined with a current error feedback algorithm that performs an approximate feedback linearization relative to one of the input-output pairs.

9.6.1 GMAW Model

In [171] the following simplified model of the gas-metal arc welding process (GMAW) is given:

$$\begin{aligned}\dot{x}_1 &= x_2 \\ \dot{x}_2 &= \frac{1}{m(t)}(-2.5x_1 - 10^{-5}x_2 + F + I^2) \\ \dot{m} &= k_2I + k_5I^2L_s\end{aligned}$$

Reset condition:

$$m(t) = \begin{cases} m(t) & \text{if } m \leq 25 \\ m(1 - 0.8(\frac{1}{1+e^{-10x_2}}) + 0.125) & \text{otherwise} \end{cases}$$

Here x_1 denotes the position of a droplet of molten metal growing on the end of the welding wire, x_2 is its velocity, and the mass of the droplet is m . The system is forced by $F = 1$, which represents various gravitational and aerodynamic forces, and two other inputs: I , which is the current from the welding power supply, and L_s , which denotes the distance the wire protrudes from the contact tube of the welding system (simply called stickout). The other feature of the model is the “reset” condition, reflecting the fact that after the droplet grows to a certain size it detaches. In this example we simply use a fixed constraint for detachment, although in reality the condition is variable. After detachment some mass remains attached to the end of the wire, resulting in a new droplet that begins growing until it detaches. The amount of mass that remains behind is a key to the dynamic behavior of the process. In the model above we have made the mass that detaches proportional to a sigmoidal function of droplet velocity. We also reset the velocity and position to zero whenever the mass detaches. This produces a model that in fact closely emulates a real GMAW process in what is called the globular mode. Figure 9.12 gives a typical uncontrolled response of the mass for this model, where $I = 1$ and $L_s = 0.1$.

9.6.2 ILC-Based Control Strategy

As noted above, our control objective is to force the mass to detach at regular intervals with a uniform amount of mass in each detached droplet. This implies the desired waveform for the mass of the droplet as shown in Figure 9.13. This waveform has a maximum equal to the value at which the GMAW model resets. This is a simplification that we will relax in future work.

In considering ways to control the GMAW process, one idea was to consider a droplet detachment as an event. This in turn led to the idea of

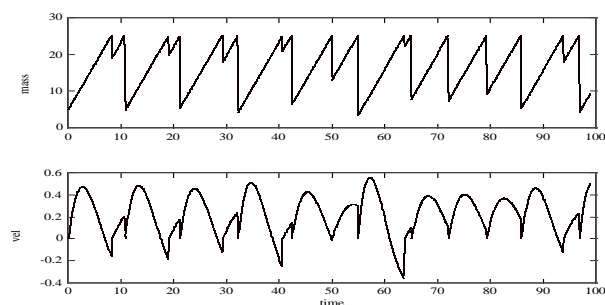


FIGURE 9.12. Uncontrolled response of the GMAW model.

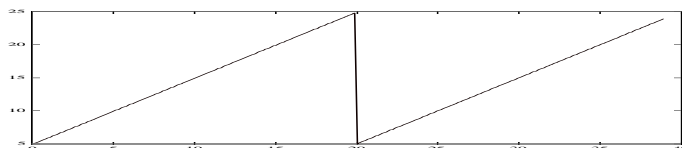


FIGURE 9.13. Desired mass waveform.

viewing each detachment event as a trial, which then motivated us to consider an ILC approach. Specifically, the time interval between detachments of mass droplets from the end of the consumable electrode is considered to be a trial. Because this interval and the amount of mass that detaches is deterministically unpredictable for some operating points one would think that we could not use ILC because of having violated the basic ILC assumptions of requiring both the trial length and the initial conditions at the beginning of each trial to be uniform. Nevertheless, as we show below it is possible to control the process to follow the desired mass waveform using an ILC approach.

A standard ILC-type algorithm for updating the control inputs for the GMAW system dynamics without the reset condition, and with a fixed trial

length might have the form:

$$\begin{aligned} I_{k+1}(t) &= I_k(t) + k_p e_k(t+1) \\ L_{s_{k+1}}(t) &= L_{s_k}(t) + k_d \dot{e}_k(t+1) \end{aligned}$$

where k denotes the trial and $e = m_d(t) - m(t)$ is the mass error. The difficulty with direct application of such an algorithm in the case of the GMAW system is that in general each trial (i.e., time between detachments) may have a different length. A second difficulty is that the mass may not always reset to the same value. A third difficulty is that the system is unstable. These problems are addressed separately in our ILC algorithm:

1. *Unstable Dynamics*: First, the fact that our system is unstable leads us to introduce what is often called current cycle feedback [23]. For this system we assume both mass and velocity can be measured. We then use an approximate feedback linearization controller to control the droplet velocity by adjusting the current. This controller compensates for the division by $m(t)$ in the velocity dynamics by multiplying the current by the square root of the mass (using physical information about the process that allows us to know the functional form through which current influences the dynamics). Notice that it is necessary with this model to assume measurement of velocity. A scheme based only on mass will not work because the velocity state is not observable from mass measurements.
2. *Non-uniform reset value*: Again using knowledge of the physics of the process, at each trial we use a simple ILC routine to adjust the set-point of velocity based on the error in the reset value at the beginning of the trial.
3. *Non-uniform trial length*: There are four cases that might arise in trying to control the detachment interval in the GMAW system:
 - (a) The actual mass resets before the desired waveform resets.
 - (b) The actual mass resets after the desired waveform resets.
 - (c) Both the actual and the desired mass waveform reset simultaneously.
 - (d) Neither the actual or the desired mass waveform have reset.

Here the term reset refers to a trial completing (i.e., the mass drops off and the system resets). Space limitations prohibit a complete explanation of the reasoning behind the approach we have developed to handle the non-uniform trial length. Basically, the approach is as follows:

- (a) If the actual mass resets before the desired mass, then reset the desired mass and continue, using a typical ILC update algorithm.
- (b) If the desired mass resets before the actual mass, then (1) set the inputs and the desired mass to some fixed waveform (e.g., a nominal constant); (2) continue until the actual mass resets; (3) reset the desired mass when the actual mass resets and continue using a typical ILC update algorithm.
- (c) If the actual mass on previous trials has always reset at a time less than the desired reset time, then the first time the system goes past the longest time the system had ever previously run the ILC algorithm will not have an error signal to use. To handle this all errors should be initialized at zero and updated only as data becomes available.

What is happening in item (b) is that if the actual system has not reset by the time we want it to, we simply suspend the ILC algorithm. That is, it should not be doing anything at time greater than the period of the desired output waveform. Thus, we just provide the system a nominal input and wait until the actual mass resets. We then continue, computing the errors for our ILC algorithm from the beginning of the past trial. Likewise, in item (c) we wish to ensure that, if the duration of the previous trial was shorter than desired, but the current trial's length is longer than the previous trial's length, then we do not compute changes to the input for times greater than the previous trial (because then you would actually be in the current trial).

4. *ILC algorithm to adjust the slope*: The final piece of our controller is the use of a standard ILC algorithm to update the stickout based on errors from the previous trial. We do not adjust the current, which is dedicated to controlling the velocity using current (present) error feedback.

Algorithmically, the ILC procedure can be represented in terms of a new quantity, called the "duration." The duration, denoted as t_{dk} , is the length of trial k . Using this notation, and defining the desired mass to be $m_d(t)$, the desired trial length to be t_{dd} , and the starting time of each trial to be t_{sk} , the ILC algorithm can be written as:

$$\begin{aligned}\bar{I}(t) &= (|\bar{I}^2(t-1) - k_1(Vsp_k - v(t-1)) \\ &\quad + k_2(Vsp_k - v(t))|)^{1/2} \\ Vsp_k &= Vsp_{k-1} - k_3(m_d(t_{sk})) - m(t_{sk})\end{aligned}$$

$$I(t) = \bar{I}(t)(m(t))^{1/2}$$

$$L_s(t) = \begin{cases} L_s(t - t_{dk-1}) \\ \quad + k_4(\dot{m}_d(t - t_{dk-1} + 1) \\ \quad \quad - \dot{m}(t - t_{dk-1} + 1)) & \text{if } (t - t_{sk}) \\ \quad \quad \quad \leq t_{dk-1} \\ L_s(t - 1) & \text{otherwise} \end{cases}$$

It should be emphasized again that the desired mass waveform is defined according to when the actual trial ends relative to the desired trial length. If the actual trial ends then the desired mass is reset to its initial value as the next trial begins. If the trial lasts longer than the desired length then the desired mass is set as follows (and, the ILC algorithm for stickout is discontinued until the next trial begins):

$$m_d(t) = \begin{cases} m_d(t - t_{sk}) & \text{if } (t - t_{sk}) \leq t_{dd} \\ m_{d\max} & \text{otherwise} \end{cases}$$

Figure 9.14 shows a sample simulation of the ILC algorithm applied to the same open-loop system shown in Figure 9.12. It can be seen that the frequency of the system is locked into the desired frequency within less than ten trials (detachments events). Note that for our algorithm and desired mass waveform it is essential that both the error between the actual and desired mass waveform and the derivative of the error go to zero. This is because we are resetting the desired waveform to its initial value each time a detachment event occurs.

The results we have presented here are very promising, especially from a theoretical perspective. The idea of a variable trial is novel in the area of iterative learning control, as is the idea of a variable initial condition at the beginning of each trial. There are a number of things that we are planning as a follow-on to this work. We are currently working to relax the assumption of a fixed reset criteria. We have also begun to develop a theoretical explanation for the effectiveness of our algorithms. This includes establishing the class of systems to which the technique can be applied as well as studying convergence. Finally, we are considering application of these ideas to develop the notion of a generalized phase-locked, frequency-locked loop for nonlinear systems.

9.7 Conclusion: The Past, Present, and Future of ILC

In this paper we have given an overview of the field of iterative learning control. We have illustrated the essential nature of the approach using several examples, discussions of descriptive figures, and through a discussion of the connection between ILC and other common control paradigms. We

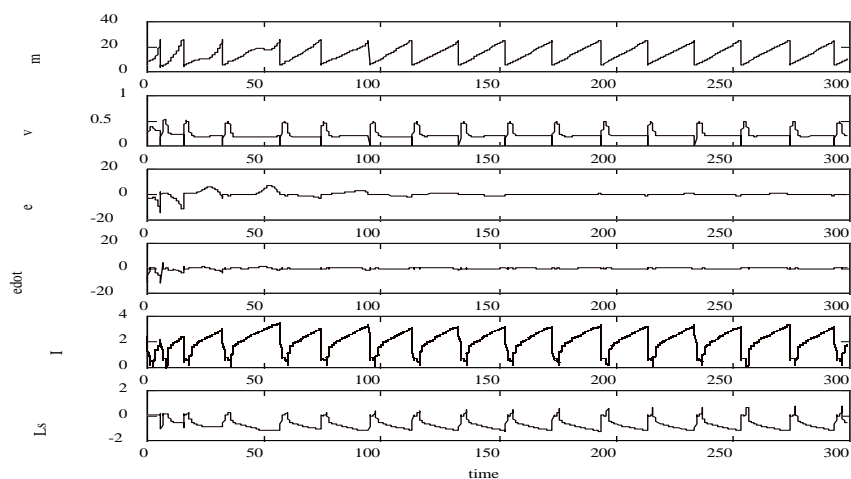


FIGURE 9.14. System response using the iterative learning control algorithm: (a) mass; (b) velocity; (c) error; (d) derivative of error; (e) current; (f) stickout.

have given a comprehensive introduction to the literature, including a topical classification of some of the literature and a summary of the major algorithms, results, applications, and emerging areas of ILC.

Because ILC as a distinct field is perhaps less than fifteen years old, it is difficult to assess its past history, its present value, and the potential future impact it may have in the world of control systems. Regarding the *past* of ILC, it is clear that the pioneering work of Arimoto and his colleagues stimulated a new approach to controlling certain types of repetitive systems. The concept of iterative learning is quite natural, but had not been expressed in the algorithmic form of ILC until the early 1980's. As we have described in this paper, early work in the field demonstrated the usefulness and applicability of the concept of ILC, particularly for linear systems, some classes of nonlinear systems, and the well-defined dynamics of robotic systems. The *present* status of the field reflects the continuing efforts of researchers to extend the earlier results to broader classes of systems, to apply these results to a wider range of applications, and to understand and inter-

pret ILC in terms of other control paradigms and in the larger context of learning in general. Looking to the *future*, it seems clear to this author that there a number of areas of research in ILC that promise to be important. These include:

1. *Integrated Higher-Order ILC/Current-Cycle Feedback*: We have noted that by including current-cycle feedback together with higher-order past-cycle feedback it is possible to simultaneously stabilize and achieve the desired performance, including possible improvements in convergence rates from the system. However, more work is needed to understand this approach.
2. *Continuous ILC/Repetitive Control*: This is one of the most important areas for future research. The last example presented above shows the value of such an approach. What is needed now is to understand how the technique can be made more general. Also it must also be reconciled with the more strict definitions of repetitive and periodic control. However, a particular vision of this author is that ILC can be used in a continuous situation when the goal is to produce a periodic response so as to produce what can be called a nonlinear, phase-locked, frequency-locked loop. This can lead to significant applications in motor control (pulse-width modulated systems) and communications.
3. *Robustness and Convergence Analysis*: What is still needed is more rigorous and more conclusive results on when algorithms will converge and how robust this convergence will be. Such information can be used to develop comprehensive theories of ILC algorithm design.
4. *System-Theoretic Analysis*: In the same vein as robustness and convergence, more work is needed to characterize the capabilities of ILC algorithms. One such analysis is extend the 2-D analysis that some authors have applied to linear systems to the nonlinear case in order to provide combined global convergence and stability results. Another is to explore the connections with other optimal control methodologies. As an example, if one poses the problem of L_2 minimization of error on a fixed interval $[0, t_f]$ and solves it using ILC, one would expect the resulting $u^*(t)$ to be the same as the $u^*(t)$ that results from solving a standard linear quadratic regulator problem on the same interval. It would also be interesting to consider the same type of comparison for mixed-sensitivity problems that do not have analytical solutions to see if it is possible in such cases to derive the optimal input using ILC. These same comments also apply to ILC for nonlinear systems.
5. *Connections to More General Learning Paradigms*: One of the important areas of research for ILC in the future will be developing

ways to make it more general and understanding ways to connect it to other learning methodologies. The ILC methodology described in this paper can be described as “trajectory learning.” Unfortunately, if a new trajectory is introduced, the algorithms typically “forget” what was learned in previous learning cycles. As we have noted, some researchers have considered this problem [112, 113, 239] but there is much more work to be done. Of particular importance is to connect ILC with some of the object-oriented approaches to learning that are being developed in the artificial intelligence community.

6. *Wider Variety of Applications:* There will almost certainly be a wider-variety of applications of ILC in the future, particularly as more results are developed related to continuous and current-cycle ILC.

In short, iterative learning control is an interesting approach to control based on the idea of iterative refinement of the input to a system based on errors recorded during past trials. It has had a stimulating period of development up to its present state-of-the art and it has a promising future, with numerous areas open for continued research and application.

Acknowledgments: This project has been partially funded by the INEEL University Research Consortium. The INEEL is managed by Lockheed Martin Idaho Technologies Company for the U.S. Department of Energy, Idaho Operations Office, under Contract No. DE-AC07-94ID13223. The author also gratefully acknowledges Ms. Anna Mathews’ assistance in procuring references through the Idaho State University interlibrary loan system.

- [1] Hyun-Sik Ahn, Soo-Hyung Lee, and Do-Hyun Kim. Frequency-domain design of iterative learning controllers for feedback systems. In *IEEE International Symposium on Industrial Electronics*, volume 1, pages 352–357, Athens, Greece, July 1995.
- [2] J. Albus. Outline for a theory of intelligence. *IEEE Transactions on Systems, Man, and Cybernetics*, 21(3):473–509, May/June 1991.
- [3] David M. Alter and Tsu-Chin Tsao. Two-dimensional exact model matching with application to repetitive control. *Journal of Dynamic Systems, Measurement, and Control*, 116:2–9, March 1994.
- [4] N. Amann and D. H. Owens. Non-minimum phase plants in iterative learning control. In *Second International Conference on Intelligent Systems Engineering*, pages 107–112, Hamburg-Harburg, Germany, September 1994.
- [5] N. Amann and D. H. Owens. Iterative learning control for discrete time systems using optimal feedback and feedforward actions. In

- Proceedings of the 34th Conference on Decision and Control*, New Orleans, LA, July 1995.
- [6] N. Amann, D. H. Owens, and E. Rogers. 2D systems theory applied to learning control systems. In *Proceedings of the 33rd Conference on Decision and Control*, Lake Buena Vista, FL, December 1994.
- [7] N. Amann, D. H. Owens, and E. Rogers. Iterative learning control using optimal feedback and feedforward actions. *International Journal of Control*, 65(2):277–293, September 1996.
- [8] N. Amann, D. H. Owens, and E. Rogers. Robustness of norm-optimal iterative learning control. In *Proceedings of International Conference on Control*, volume 2, pages 1119–1124, Exeter, UK, September 1996.
- [9] N. Amann, D. H. Owens, E. Rogers, and A. Wahl. An H_∞ approach to linear iterative learning control design. *International Journal of Adaptive Control and Signal Processing*, 10(6):767–781, November–December 1996.
- [10] N. Amann, H. Owens, and E. Rogers. Iterative learning control for discrete-time systems with exponential rate of convergence. *IEEE Proceedings-Control Theory and Applications*, 143(2):217–224, March 1996.
- [11] S. Arimoto. Mathematical theory of learning with applications to robot control. In *Proceedings of 4th Yale Workshop on Applications of Adaptive Systems*, pages 379–388, New Haven, Connecticut, May 1985.
- [12] S. Arimoto. System theoretic study of learning control. *Transactions of Society of Instrumentation and Control Engineers*, 21(5):445–450, 1985.
- [13] S. Arimoto, S. Kawamura, and F. Miyazaki. Can mechanical robots learn by themselves? In *Proceedings of 2nd International Symposium of Robotics Research*, pages 127–134, Kyoto, Japan, August 1984.
- [14] S. Arimoto, S. Kawamura, and F. Miyazaki. Bettering operation of dynamic systems by learning: A new control theory for servomechanism or mechatronic systems. In *Proceedings of 23rd Conference on Decision and Control*, pages 1064–1069, Las Vegas, Nevada, December 1984.
- [15] S. Arimoto, S. Kawamura, and F. Miyazaki. Bettering operation of robots by learning. *Journal of Robotic Systems*, 1(2):123–140, March 1984.

- [16] S. Arimoto, S. Kawamura, and F. Miyazaki. Iterative learning control for robot systems. In *Proceedings of IECON*, Tokyo, Japan, October 1984.
- [17] S. Arimoto, S. Kawamura, F. Miyazaki, and S. Tamaki. Learning control theory for dynamic systems. In *Proceedings of 24th Conference on Decision and Control*, pages 1375–1380, Ft. Lauderdale, Florida, December 1985.
- [18] Suguru Arimoto. Ability of motion learning comes from passivity and dissipativity of its dynamics. In *Proceedings of the 2nd Asian Control Conference*, Seoul, Korea, July 1997.
- [19] C. Atkeson and J. McIntyre. Robot trajectory learning through practice. In *Proceedings of the 1986 IEEE Conference on Robotics and Automation*, San Francisco, California, April 1986.
- [20] A. Barto, R. Sutton, and C. Anderson. Neuron-like adaptive elements that can solve difficult learning control problems. *IEEE Transactions on Systems, Man, and Cybernetics*, 13(5):834–846, September 1983.
- [21] Z. Bien and K. M. Huh. Higher-order iterative control algorithm. In *IEE Proceedings Part D, Control Theory and Applications*, volume 136, pages 105–112, May 1989.
- [22] Z. Bien, D. Hwang, and S. Oh. A nonlinear iterative learning method for robot path control. *Robotica*, 9:387–392, 1991.
- [23] Shun Boming. *On Iterative Learning Control*. PhD thesis, National University of Singapore, Singapore, 1996.
- [24] P. Bondi, G. Casalino, and L. Gambardella. On the iterative learning control theory for robotic manipulators. *IEEE Journal of Robotics and Automation*, 4(1):14–22, February 1988.
- [25] Gary M. Bone. A novel iterative learning control formulation of generalized predictive control. *Automatica*, 31(10):1483–1487, 1995.
- [26] C. K. Chang, R. W. Longman, and M. Phan. Techniques for improving transients in learning control systems. *Advances in the Astronautical Sciences, American Astronautical Society*, 76:2035–2052, 1992.
- [27] W.S. Chang and I.H. Suh. Analysis and design of dual-repetitive controllers. In *Proceedings of the 35th IEEE Conference on Decision and Control*, Kobe, Japan, December 1996.
- [28] Chien Chern Cheah and Danwei Wang. Model reference learning control scheme for a class of nonlinear systems. *International Journal of Control*, 66(2):271–287, January 1997.

- [29] Chien Chong Chen, Chyi Hwang, and Ray Y. K. Yang. Optimal periodic forcing of nonlinear chemical processes for performance improvements. *The Canadian Journal of Chemical Engineering*, 72:672–682, August 1994.
- [30] Peter C. Y. Chen, James K. Mills, and Kenneth C. Smith. Performance improvement of robot continuous-path operation through iterative learning using neural networks. *Machine Learning*, 23:191–220, May-June 1996.
- [31] Y. Chen and H. Dou. Robust curve identification by iterative learning. In *Proceedings of the 1st Chinese World Congress on Intelligent Control and Intelligent Automation*, Beijing, China, 1993.
- [32] Y. Chen, M. Sun, and H. Dou. Dual-staged P-type iterative learning control schemes. In *Proceedings of the 1st Asian Control Conference*, Tokyo, Japan, 1994.
- [33] Y. Chen, C. Wen, and M. Sun. Discrete-time iterative learning control of uncertain nonlinear feedback systems. In *Proceedings of the 2nd Chinese World Congress on Intelligent Control and Intelligent Automation*, Xi'an, China, 1997.
- [34] Y. Chen, C. Wen, and M. Sun. A high-order iterative learning controller with initial state learning. In *Proceedings of the 2nd Chinese World Congress on Intelligent Control and Intelligent Automation*, Xi'an, China, 1997.
- [35] Y. Chen, C. Wen, J.-X. Xu, and M. Sun. extracting projectile's aerodynamic drag coefficient curve via high-order iterative learning identification. In *Proceedings of 35th IEEE Conference on Decision and Control*, Kobe, Japan, December 1996.
- [36] Y. Chen, J.-X. Xu, and T.H. Lee. Feedback-assisted high-order iterative learning control of uncertain nonlinear discrete-time systems. In *Proceedings of the International Conference on Control, Automation, Robotics, and Vision*, Singapore, December 1996.
- [37] Y. Chen, J.-X. Xu, T.H. Lee, and S. Yamamoto. Comparative studies of iterative learning control schemes for a batch chemical process. In *Proceedings of the IEEE Singapore Int. Symposium on Control Theory and Applications*, Singapore, July 1997.
- [38] Y. Chen, J.-X. Xu, T.H. Lee, and S. Yamamoto. An iterative learning control in rapid thermal processing. In *Proceedings of IASTED Int. Conference on Modeling, Simulation, and Optimization*, Singapore, August 1997.

- [39] Y. Chen, J.-X. Xu, and T.-H. Tong. An iterative learning controller using current iteration tracking error information and initial state learning. In *Proceedings of the 35th IEEE Conference on Decision and Control*, Kobe, Japan, December 1996.
- [40] Yangquan Chen, Changyun Wen, Jian-Xin Xu, and Mingxuan Sun. Initial state learning method for iterative learning control of uncertain time-varying systems. In *Proceedings of the 35th IEEE Conference on Decision and Control*, volume 4, pages 3996–4001, Kobe, Japan, December 1996.
- [41] Yangquan Chen, Jian-Xin Xu, and Tong Heng Lee. Current iteration tracking error assisted iterative learning control of uncertain nonlinear discrete-time systems. In *Proceedings of the 35th IEEE Conference on Decision and Control*, volume 3, pages 3038–3043, Kobe, Japan, December 1996.
- [42] Yangquan Chen, Jian-Xin Xu, and tong Heng Lee. Current iteration tracking error assisted high-order iterative learning control of discrete-time uncertain nonlinear systems. In *Proceedings of the 2nd Asian Control Conference*, Seoul, Korea, July 1997.
- [43] Kok Kia Chew and Masayoshi Tomizuka. Steady state and stochastic performance of a modified discrete-time prototype repetitive controller. *Journal of Dynamic Systems, Measurement, and Control*, 112:35–41, March 1990.
- [44] Chiang-Ju Chien. A discrete iterative learning control of nonlinear time-varying systems. In *Proceedings of the 35th IEEE Conference on Decision and Control*, volume 3, pages 3056–3061, Kobe, Japan, December 1996.
- [45] Chiang-Ju Chien. On the iterative learning control of sampled-data systems. In *Proceedings of the 2nd Asian Control Conference*, Seoul, Korea, July 1997.
- [46] Chiang-Ju Chien and Jing-Sin Liu. P-type iterative learning controller for robust output tracking of nonlinear time-varying systems. *International Journal of Control*, 64(2):319–334, May 1996.
- [47] Tsu chin Tsao and Masayoshi Tomizuka. Robust adaptive and repetitive digital tracking control and application to a hydraulic servo for noncircular machining. *Journal of Dynamic Systems, Measurement, and Control*, 116:24–32, March 1994.
- [48] Chong-Ho Choi and Tae-Jeong Jang. Iterative learning control for a general class of nonlinear feedback systems. In *Proceedings of the 1995 American Control Conference*, pages 2444–2448, Seattle, WA, June 1995.

- [49] Chong-Ho Choi and Tae-Jeong Jang. Iterative learning control of nonlinear systems with consideration on input energy. In *Proceedings of the 2nd Asian Control Conference*, Seoul, Korea, July 1997.
- [50] Jeong-Woo Choi, Hyun-Goo Choi, Kwang-Soon Lee, and Won-Hong Lee. Control of ethanol concentration in a fed-batch cultivation of *acinetobacter calcoaceticus* RAG-1 using a feedback-assisted iterative learning algorithm. *Journal of Biotechnology*, 49:29–43, August 1996.
- [51] Jin Young Choi and Hyun Joo Park. Neural-based iterative learning control for unknown systems. In *Proceedings of the 2nd Asian Control Conference*, Seoul, Korea, July 1997.
- [52] Wonsu Choi and Joongsun Yoon. An intelligent time-optimal control. In *Proceedings of the 2nd Asian Control Conference*, Seoul, Korea, July 1997.
- [53] Michel Cotsaftis. Robust asymptotic stability for intelligent controlled dynamical systems. In *Proceedings of the 2nd Asian Control Conference*, Seoul, Korea, July 1997.
- [54] J. Craig, P. Hsu, and S. Sastry. Adaptive control of mechanical manipulators. In *Proceedings of IEEE Conference on Robotics and Automation*, pages 190–195, San Francisco, California, April 1986.
- [55] J.J. Craig. Adaptive control of manipulators through repeated trials. In *Proceedings of 1984 American Control Conference*, pages 1566–1572, San Diego, California, June 1984.
- [56] John Craig. *Adaptive Control of Robot Manipulators*. Addison-Wesley, Reading, Massachusetts, 1988.
- [57] Y.Q. Dai, K. Usui, and M. Uchiyama. A new iterative approach to solving inverse flexible-link manipulator kinematics. In *Proceedings of the 35th IEEE Conference on Decision and Control*, Kobe, Japan, December 1996.
- [58] D. de Roover. Synthesis of a robust iterative learning controller using an H_∞ approach. In *Proceedings of the 35th IEEE Conference on Decision and Control*, Kobe, Japan, December 1996.
- [59] Alessandro DeLuca and Stefano Panzieri. End-effector regulation of robots with elastic elements by an iterative scheme. *International Journal of Adaptive Control and Signal Processing*, 10:379–393, July–October 1996.
- [60] Zhidong Deng, Zaixing Zhang, and Zengqi Sun. Asynchronous stochastic learning control with accelerative factor in multivariable systems. In *Proceedings of the 1995 IEEE International Conference*

- on Systems, Man, and Cybernetics*, volume 4, pages 3790–3794, Vancouver, BC, Canada, October 1995.
- [61] Tae-Yong Doh, Jung-Hoo Moon, Kyung Bog Jin, and Myung Jin Chung. An iterative learning control for uncertain systems using structured singular value. In *Proceedings of the 2nd Asian Control Conference*, Seoul, Korea, July 1997.
- [62] H. Dou, Z. Zhou, Y. Chen, J.-X. Xu, and J. Abbas. Robust motion control of electrically stimulated human limb via discrete-time high-order iterative learning scheme. In *Proceedings of the International Conference on Control, Automation, Robotics, and Vision*, Singapore, December 1996.
- [63] Huifang Dou, Zhaoying Zhou, Yangquan Chen, Jian-Xin Xu, and James J. Abbas. Robust control of functional neuromuscular stimulation system by discrete-time iterative learning scheme. In *Proceedings of the 2nd Asian Control Conference*, Seoul, Korea, July 1997.
- [64] Huifang Dou, Zhaoying Zhou, Mingxuan Sun, and Yangquan Chen. Robust high-order P-type iterative learning control for a class of uncertain nonlinear systems. In *Proceedings of the IEEE International Conference on Systems, Man, and Cybernetics*, volume 2, pages 923–928, Beijing, China, October 1996.
- [65] H. Elci, R.W. Longman, M.Q. Phan, J. Juang, and R. Ugoletti. Automated learning control through model updating for precision motion control. In *ASME Winter Meeting, Adaptive Structures and Materials Systems Symposium*, Chicago, Illinois, November 1994.
- [66] H. Elci, R.W. Longman, M.Q. Phan, J. Juang, and R. Ugoletti. Discrete frequency-based learning control for precision motion control. In *Proceedings of the 1994 IEEE Conference on Systems, Man, and Cybernetics*, San Antonio, Texas, October 1994.
- [67] H. Elci, M. Phan, and R. W. Longman. Experiments in the use of learning control for maximum precision robot trajectory tracking. In *Proceedings of the 1994 Conference on Information Science and Systems*, pages 951–958, Princeton University, Department of Electrical Engineering, Princeton, NJ, 1994.
- [68] H. Elci, H. N. Juang R. W. Longman, M. Phan, and R. Ugoletti. Automated learning control through model updating for precision motion control. *Adaptive Structures and Composite Materials: Analysis and Applications*, ASME, AD-45/MD-54:299–314, 1994.
- [69] J. A. Franklin. Refinement of robot motor skills through reinforcement learning. In *Proceedings of 27th Conference on Decision and Control*, pages 1096–1101, Austin, Texas, December 1988.

- [70] James A. Frueh and Minh Q. Phan. System identification and inverse control using input-output data from repetitive trials. In *Proceedings of the 2nd Asian Control Conference*, Seoul, Korea, July 1997.
- [71] K.S. Fu. Learning control systems — review and outlook. *IEEE Transactions on Automatic Control*, 15:210–215, April 1970.
- [72] Mitsuo Fukuda and Seiichi Shin. Model reference learning control with a wavelet network. In *Proceedings of the 2nd Asian Control Conference*, Seoul, Korea, July 1997.
- [73] K. Furuta and M. Yamakita. The design of a learning control system for multivariable systems. In *Proceedings of IEEE International Symposium on Intelligent Control*, pages 371–376, Philadelphia, Pennsylvania, January 1987.
- [74] S. S. Garimella and K. Srinivasan. Application of iterative learning control to coil-to-coil control in rolling. In *Proceedings of the 1995 American Control Conference*, Seattle, WA, June 1995.
- [75] Zheng Geng, Robert Carroll, and Jahong Xie. Two-dimensional model and algorithm analysis for a class of iterative learning control systems. *International Journal of Control*, 52(4):833–862, December 1989.
- [76] D. Gorinevsky. An approach to parametric nonlinear least square optimization and application to task-level learning control. *IEEE Transactions on Automatic Control*, 42(7):912–927, 1997.
- [77] Y.L. Gu and N.K. Loh. Learning control in robotic systems. In *Proceedings of IEEE International Symposium on Intelligent Control*, pages 360–364, Philadelphia, Pennsylvania, January 1987.
- [78] Kennon Guglielmo and Nader Sadegh. Theory and implementation of a repetitive robot controller with Cartesian trajectory description. *Journal of Dynamic Systems, Measurement, and Control*, 118:15–21, March 1996.
- [79] Reed D. Hanson and Tsu-Chin Tsao. Compensation for cutting force induced bore cylindricity dynamic errors - a hybrid repetitive servo/iterative learning process feedback control approach. In *Proceedings of the Japan/USA Symposium on Flexible Automation*, volume 2, pages 1019–1026, Boston, MA, July 1996.
- [80] S. Hara, T. Omata, and M. Nakano. Synthesis of repetitive control systems and its application. In *Proceedings of 24th Conference on Decision and Control*, pages 1387–1392, Ft. Lauderdale, Florida, December 1985.

- [81] S. Hara, Y. Yamamoto, T. Omata, and M. Nakano. Repetitive control system: A new type servo system for periodic exogenous signals. *IEEE Transactions on Automatic Control*, 33(7):659–668, July 1988.
- [82] E. Harokopos. Optimal learning control of mechanical manipulators in repetitive motions. In *Proceedings of IEEE International Symposium on Intelligent Control*, pages 396–401, Philadelphia, Pennsylvania, January 1987.
- [83] H. Hashimoto and J.-X. Xu. Learning control systems with feedback. In *Proceedings of the IEEE Asian Electronics Conference*, Hong Kong, September 1987.
- [84] J.E. Hauser. Learning control for a class of nonlinear systems. In *Proceedings of 26th Conference on Decision and Control*, pages 859–860, Los Angeles, California, December 1987.
- [85] G. Heinzinger, D. Fenwick, B. Paden, and F. Miyazaki. Robust learning control. In *Proceedings of the 28th Conference on Decision and Control*, pages 436–440, Tampa, Florida, December 1989.
- [86] Greg Heinzinger, Dan Fenwick, Brad Paden, and Fumio Miyazaki. Stability of learning control with disturbances and uncertain initial conditions. *IEEE Transactions on Automatic Control*, 37(1), January 1992.
- [87] L. Hideg and R. Judd. Frequency domain analysis of learning systems. In *Proceedings of the 27th Conference on Decision and Control*, pages 586–591, Austin, Texas, December 1988.
- [88] L. M. Hideg. Stability and convergence issues in iterative learning control. In *Proceedings Intelligent Engineering Systems Through Artificial Neural Networks in Engineering Conference*, volume 4, pages 211–216, St. Louis, MO, November 1994.
- [89] L. M. Hideg. Stability of LTV MIMO continuous-time learning control schemes: a sufficient condition. In *Proceedings of the 1994 IEEE International Symposium on Intelligent Control*, pages 285–290, 1995.
- [90] L. M. Hideg. Time delays in iterative learning control schemes. In *Proceedings of the 1995 IEEE International Symposium on Intelligent Control*, pages 5–20, Monterey, CA, August 1995.
- [91] L. M. Hideg. Stability and convergence issues in iterative learning control - II. In *Proceedings of the 1996 IEEE International Symposium on Intelligent Control*, pages 480–485, Dearborn, MI, September 1996.

- [92] Gunnar Hillerstrom. Adaptive suppression of vibrations—a repetitive control approach. *IEEE Transactions on Control Systems Technology*, 4(1), January 1996.
- [93] Gunnar Hillerstrom and Jan Sternby. Application of repetitive control to a peristaltic pump. *Journal of Dynamic Systems, Measurement, and Control*, 116:786–789, December 1994.
- [94] J. Hodgson. Structures for intelligent systems. In *Proceedings of IEEE International Symposium on Intelligent Control*, pages 348–353, Philadelphia, Pennsylvania, January 1987.
- [95] R. Horowitz, W. Messner, and J. B. Moore. Exponential convergence of a learning controller for robot manipulators. *IEEE Transactions on Automatic Control*, 36(7):890–894, July 1991.
- [96] Roberto Horowitz. Learning control of robot manipulators. *Journal of Dynamic Systems, Measurement, and Control*, 115:402–411, June 1993.
- [97] Jwusheng Hu and Masayoshi Tomizuka. A digital semented repetitive control algorithm. *Journal of Dynamic Systems, Measurement, and Control*, 116:577–582, December 1994.
- [98] Jwushung Hu and Masayoshi Tomizuka. Adaptive asymptotic tracking of repetitive signals - a frequency domain approach. *IEEE Transactions on Automatic Control*, 38(19), October 1993.
- [99] Y. C. Huang and R. W. Longman. The source of the often observed property of initial convergence followed by divergence in learning and repetitive control. *Advances in the Astronautical Sciences*, 90:555–572, 1996.
- [100] Kyungmoo Huh. A study on the robustness of the higher-order iterative learning control algorithm. In *Proceedings of the 2nd Asian Control Conference*, Seoul, Korea, July 1997.
- [101] D. H. Hwang, Z. Bien, and S. R. Oh. Iterative learning control method for discrete-time dynamic systems. In *IEE Proceedings Part D, Control Theory and Applications*, volume 138, pages 139–144, March 1991.
- [102] 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*, pages 314–323, Coordinated Sciences Laboratory, University of Illinois at Urbana-Champaign, 1994.

- [103] H. S. Jang and R. W. Longman. Design of digital learning controllers using a partial isometry. *Advances in the Astronautical Sciences*, 93:137–152, 1996.
- [104] Tae-Jeong Jang, Chong-Ho Choi, and Hyun-Sik Ahn. Iterative learning control feedback systems. *Automatica*, 31(2):243–248, February 1995.
- [105] Doyoung Jeon and Masayoshi Tomizuka. Learning hybrid force and position control of robot manipulators. *IEEE Transactions on Robotics and Automation*, 9:423–431, August 1993.
- [106] Y. An Jiang, D. J. Clements, and T. Hesketh. Betterment learning control of nonlinear systems. In *Proceedings of the 1995 IEEE Conference on Decision and Control*, volume 2, pages 1702–1707, New Orleans, LA, December 1995.
- [107] W. Jouse and J. Williams. PWR Heat-Up Control by Means of a Self-Teaching Neural Network. In *Proceedings of International Conference on Control and Instrumentation in Nuclear Installations*, Glasgow, May 1990.
- [108] R. P. Judd, R. P. Van Til, and L. Hideg. Equivalent Lyapunov and frequency domain stability conditions for iterative learning control systems. In *Proceedings of 8th IEEE International Symposium on Intelligent Control*, pages 487–492, 1993.
- [109] S. Kawamura, F. Miyazaki, and S. Arimoto. Applications of learning method for dynamic control of robot manipulators. In *Proceedings of 24th Conference on Decision and Control*, pages 1381–1386, Ft. Lauderdale, Florida, December 1985.
- [110] S. Kawamura, F. Miyazaki, and S. Arimoto. Hybrid position/force control of robot manipulators based on learning method. In *Proceedings of '85 International Conference on Advanced Robotics*, pages 235–242, Tokyo, Japan, September 1985.
- [111] S. Kawamura, F. Miyazaki, and S. Arimoto. Convergence, stability and robustness of learning control schemes for robot manipulators. In *Proceedings of the International Symposium on Robotic Manipulators: Modelling, Control, and Education*, Albuquerque, New Mexico, November 1986.
- [112] S. Kawamura, F. Miyazaki, and S. Arimoto. Intelligent control of robot motion based on learning method. In *Proceedings of IEEE International Symposium on Intelligent Control*, pages 365–370, Philadelphia, Pennsylvania, January 1987.

- [113] S. Kawamura, F. Miyazaki, and S. Arimoto. Realization of robot motion based on a learning method. *IEEE Transactions on Systems, Man, and Cybernetics*, 18(1):126–134, January/February 1988.
- [114] S. Kawamura, F. Miyazaki, M. Matsumori, and S. Arimoto. Learning control scheme for a class of robot systems with elasticity. In *Proceedings of 25th Conference on Decision and Control*, pages 74–79, Athens, Greece, December 1986.
- [115] Sadao Kawamura and Norihisa Fukao. Use of feedforward input patterns obtained through learning control. In *Proceedings of the 2nd Asian Control Conference*, Seoul, Korea, July 1997.
- [116] M. Kawato, K. Furukawa, and R. Suzuki. A hierarchical neural-network model for control and learning of voluntary movement. *Biological Cybernetics*, 57:169–185, February 1987.
- [117] B. K. Kim, W. K. Chung, and Y. Youm. Robust learning control for robot manipulators based on disturbance observer. In *IECON Proceedings (Industrial Electronics Conference)*, volume 2, pages 1276–1282, Taipei, Taiwan, 1996.
- [118] Bong-Keun Kim, Wan-Kyun Chung, and Youngil Youm. Iterative learning method for trajectory control of robot manipulators based on disturbance observer. In *Proceedings of the 2nd Asian Control Conference*, Seoul, Korea, July 1997.
- [119] Dong-Il Kim and Sungkwun Kim. On iterative learning control algorithm for industrial robots and CNC machine tools. In *Proceedings of IECON '93 - 19th Annual Conference of IEEE Industrial Electronics*, volume 1, pages 601–606, Maui, HI, November 1993.
- [120] Dong-Il Kim and Sungkwun Kim. An iterative learning control method with application for CNC machine tools. *IEEE Transactions on Industry Applications*, 32(1):66–72, January-February 1996.
- [121] Won Cheol Kim and Kwang Soon Lee. Design of quadratic criterion-based iterative learning control by principal component analysis. In *Proceedings of the 2nd Asian Control Conference*, Seoul, Korea, July 1997.
- [122] Jacek B. Krawczyk. Properties of repetitive control of partially observed processes. *Computers and Mathematics with Applications*, 24:111–124, October-November 1992.
- [123] Tae-Yong Kuc and Chae-Wook Chung. Learning control of a class of hybrid nonlinear systems. In *Proceedings of the 2nd Asian Control Conference*, Seoul, Korea, July 1997.

- [124] Tae-Yong Kuc and Chae-Wook Chung. An optimal learning control of robot manipulators. In *Proceedings of the 2nd Asian Control Conference*, Seoul, Korea, July 1997.
- [125] Tae-Yong Kuc, Jin S. Lee, and Byung-Hyun Park. Tuning convergence rate of a robust learning controller for robot manipulators. In *Proceedings of the 1995 IEEE Conference on Decision and Control*, volume 2, pages 1714–1719, New Orleans, LA, 1995.
- [126] Jerzy E. Kurek and Marek B. Zaremba. Iterative learning control synthesis based on 2-D system theory. *IEEE Transactions on Automatic Control*, 38(1), January 1993.
- [127] R. J. LaCarna and J. R. Johnson. A learning controller for the megawatt load-frequency control problem. *IEEE Transactions on Systems, Man, and Cybernetics*, 10(1):43–49, January 1980.
- [128] G. F. Ledwich and A. Bolton. Repetitive and periodic controller design. In *IEE Proceedings Part D, Control Theory and Applications*, volume 140, pages 19–24, January 1993.
- [129] Hak-Sung Lee and Zeungnam Bien. Study on robustness of iterative learning control with non-zero initial error. *International Journal of Control*, 64:345–359, June 1996.
- [130] Hak-Sung Lee and Zeungnam Bien. Robustness and convergence of a pd-type iterative learning controller. In *Proceedings of the 2nd Asian Control Conference*, Seoul, Korea, July 1997.
- [131] Ju Jang Lee and Jong Woon Lee. Design of iterative learning controller with VCR servo system. *IEEE Transactions on Consumer Electronics*, 39:13–24, February 1993.
- [132] K. S. Lee, S. H. Bang, S. Yi, J. S. Son, and S. C. Yoon. Iterative learning control of heat up phase for a batch polymerization reactor. *Journal of Process Control*, 6(4):255–262, August 1996.
- [133] Kwang Soon Lee and Jay H. Lee. Constrained model-based predictive control combined with iterative learning for batch or repetitive processes. In *Proceedings of the 2nd Asian Control Conference*, Seoul, Korea, July 1997.
- [134] S. C. Lee, R. W. Longman, and M. Phan. Direct model reference learning and repetitive control. *Advances in the Astronautical Sciences*, 85, March 1994.
- [135] T. H. Lee, J. H. Nie, and W. K. Tan. Developments in learning control enhancements for nonlinear servomechanisms. *Mechatronics*, 5(8):919–936, December 1995.

- [136] G. Lee-Glauser, J. N. Juang, and R. W. Longman. Comparison and combination of learning controllers: computational enhancement and experiments. *Journal of Guidance, Control and Dynamics*, 19(5):1116–1123, 1996.
- [137] C. James Li, Homayoon S. M. Beigi, and Shengyi Li. Nonlinear piezo-actuator control by learning self-tuning regulator. *Journal of Dynamic Systems, Measurement, and Control*, 115:720–723, December 1993.
- [138] Guo Li, Hao Li, and Jian-Xin Xu. The study and design of the longitudinal learning control system of the automobile. In *Proceedings of the 2nd Asian Control Conference*, Seoul, Korea, July 1997.
- [139] Yieth-Jang Liang and D. P. Looze. Performance and robustness issues in iterative learning control. In *Proceedings of 32nd IEEE Conference on Decision and Control*, volume 3, pages 1990–1995, San Antonio, TX, December 1993.
- [140] R. Longman, M.Q. Phan, and J. Juang. An overview of a sequence of research developments in learning and repetitive control. In *Proceedings of the First International Conference on Motion and Vibration Control*, Yokohama, Japan, September 1992.
- [141] R. W. Longman, C. K. Chang, and M. Phan. Discrete time learning control in nonlinear systems. In *Proceedings of the 1992 AIAA/AAS Astrodynamics Conference*, pages 501–511, Hilton Head, South Carolina, 1992.
- [142] R. W. Longman and Y. C. Huang. Use of unstable repetitive control for improved tracking accuracy. *Adaptive Structures and Composite Materials: Analysis and Applications*. ASME, AD-45/MD-54:315–324, 1994.
- [143] R. W Longman and Y.S. Ryu. Indirect learning control algorithms for time-varying and time-invariant systems. In *Proceedings of the Thirtieth Annual Allerton Conference on Communication, Control and Computing*, pages 731–741, Coordinated Science Laboratory, University of Illinois at Urbana-Champaign, 1992.
- [144] R. W. Longman and Y. Wang. Phase cancellation learning control using FFT weighted frequency response identification. *Advances in the Astronautical Sciences*, 93:85–101, 1996.
- [145] Richard W. Longman. Design methodologies in learning and repetitive control. In *Proceedings of the 2nd Asian Control Conference*, Seoul, Korea, July 1997.

- [146] Alessandro De Luca, Giorgio Paesano, and Giovanni Ulivi. A frequency-domain approach to learning control: implementaion for a robot manipulator. *IEEE Transactions on Industrial Electronics*, 39(1), February 1992.
- [147] P. Lucibello. Output regulation of flexible mechanisms by learning. In *Proceedings of the 1992 Conference on Decision and Control*, Tucson, AZ, December 1992.
- [148] P. Lucibello. On trajectory tracking learning of flexible joint arms. *International Journal of Robotics and Automation*, 11(4):190–194, 1996.
- [149] Pasquale Lucibello. Repositioning control of robotic arms by learning. *IEEE Transactions on Automatic Control*, 39(8), August 1994.
- [150] Pasquale Lucibello. Output zeroing with internal stability by learning. *Automatica*, 31(11):1665–1672, November 1995.
- [151] C. C. H. Ma. Stability robustness of repetitive control systems with zero phase compensation. *Journal of Dynamic Systems, Measurement, and Control*, 112:320–324, September 1990.
- [152] C. C. H. Ma. Direct adaptive rapid tracking of short complex trajectories. *Journal of Dynamic Systems, Measurement, and Control*, 116:537–541, September 1994.
- [153] Tetsuya Manabe and Fumio Miyazaki. Learning control based on local linearization by using DFT. *Journal of Robotic Systems*, 11(2):129–141, 1994.
- [154] W. Messner, R. Horowitz, W. Kao, and M. Boals. A new adaptive learning rule. *IEEE Transactions on Automatic Control*, 36(2):188–197, February 1991.
- [155] R. H. Middleton, G. C. Goodwin, and R. W. Longman. A method for improving the dynamic accuracy of a robot performing a repetitive task. *International Journal of Robotics Research*, 8:67–74, 1989.
- [156] T. Mita. Repetitive control of mechanical systems. In *Proceedings of 1984 ATACS*, Izu, Japan, 1984.
- [157] T. Mita and E. Kato. Iterative control and its application to motion control of robot arm – a direct approach to servo-problems. In *Proceedings of 24th Conference on Decision and Control*, pages 1393–1398, Ft. Lauderdale, Florida, December 1985.
- [158] T. Mita and E. Kato. Iterative control of mechanical manipulators. In *Proceedings of 15th ISIR*, Tokyo, Japan, 1985.

- [159] Jung-Ho Moon, Moon-Noh Lee, Myung Jin Chung, Soo Yul Jung, and Dong Ho Shin. Track-following control for optical disk drives using an iterative learning scheme. *IEEE Transactions on Consumer Electronics*, 42(2):192–198, May 1996.
- [160] K.L. Moore. *Design Techniques for Transient Response Control*. PhD thesis, Texas A&M University, College Station, Texas, 1989.
- [161] K.L. Moore. A reinforcement-learning neural network for the control of nonlinear systems. In *Proceedings of 1991 American Control Conference*, Boston, Massachusetts, June 1991.
- [162] K.L. Moore. *Iterative Learning Control for Deterministic Systems*. Springer-Verlag, Advances in Industrial Control, London, U.K., 1993.
- [163] K.L. Moore, S.P. Bhattacharyya, and M. Dahleh. Some results on iterative learning. In *Proceedings of 1990 IFAC World Congress*, Tallin, Estonia, USSR, August 1990.
- [164] K.L. Moore, M. Dahleh, and S.P. Bhattacharyya. Learning control for robotics. In *Proceedings of 1988 International Conference on Communications and Control*, pages 976–987, Baton Rouge, Louisiana, October 1988.
- [165] K.L. Moore, M. Dahleh, and S.P. Bhattacharyya. Artificial neural networks for learning control. TCSP Research Report 89–011, Department of Electrical Engineering, Texas A&M University, College Station, Texas, July 1989.
- [166] K.L. Moore, M. Dahleh, and S.P. Bhattacharyya. Iterative learning for trajectory control. In *Proceedings of 28th IEEE Conference on Decision and Control*, Tampa, Florida, December 1989.
- [167] K.L. Moore, M. Dahleh, and S.P. Bhattacharyya. Some results on iterative learning control. TCSP Research Report 89–004, Department of Electrical Engineering, Texas A&M University, College Station, Texas, January 1989.
- [168] K.L. Moore, M. Dahleh, and S.P. Bhattacharyya. Adaptive gain adjustment for a learning control method for robotics. In *Proceedings of 1990 IEEE International Conference on Robotics and Automation*, Cincinnati, Ohio, May 1990.
- [169] K.L. Moore, M. Dahleh, and S.P. Bhattacharyya. Iterative learning control: A survey and new results. *Journal of Robotic Systems*, 9(5), July 1992.

- [170] K.L. Moore and Anna Mathews. Iterative learning control for systems with non-uniform trial length with applications to gas-metal arc welding. In *Proceedings of the 2nd Asian Control Conference*, Seoul, Korea, July 1997.
- [171] D.S. Naidu, K.L. Moore, R. Yender, and J. Tyler. Gas metal arc welding control: Part I – modeling and analysis. In *Proceedings of the 2nd World Congress of Nonlinear Analysts*, Athens, Greece, July 1996.
- [172] K. Narendra and M. Thathachar. Learning automata — a survey. *IEEE Transactions on Systems, Man, and Cybernetics*, 4(4):323–334, July 1974.
- [173] K. Narendra and M Thathacher. *Learning Automata*. Prentice-Hall, Englewood Cliffs, New Jersey, 1989.
- [174] S.-R. Oh, Z. Bien, and I. Suh. An iterative learning control method with application for the robot manipulator. *IEEE Journal of Robotics and Automation*, 4(5):508–514, October 1988.
- [175] Giuseppe Oriolo, Stefano Panzieri, and Giovanni Ulivi. Iterative learning controller for nonholonomic robots. In *IEEE International Conference on Robotics and Automation*, volume 3, pages 2676–2681, Minneapolis, MN, 1996.
- [176] D. H. Owens. Iterative learning control-convergence using high gain feedback. In *Proceedings of the 1992 Conference on Decision and Control*, volume 4, page 3822, Tucson, AZ, December 1992.
- [177] D. H. Owens, N. Amann, and E. Rogers. Iterative learning control-an overview of recent algorithms. *Applied Mathematics and Computer Science*, 5(3):425–438, 1995.
- [178] D. H. Owens, N. Amann, and E. Rogers. Systems structure in iterative learning control. In *Proceedings of International Conference on System Structure and Control*, pages 500–505, Nantes, France, July 1995.
- [179] H. Ozaki, C. J. Lin, T. Shimogawa, and H. Qiu. Control of mechatronic systems by learning actuator reference trajectories described by B-spline curves. In *Proceedings of the IEEE International Conference on Systems, Man, and Cybernetics*, volume 5, pages 4679–4684, Vancouver, BC, Canada, October 1995.
- [180] B. H. Park and Jin S. Lee. Continuous type fuzzy learning control for a class of nonlinear dynamics system. In *Proceedings of the 2nd Asian Control Conference*, Seoul, Korea, July 1997.

- [181] Kwang-Hyun Park, Zeungnam Bien, and Dong-Hwan Hwang. Design of an iterative learning controller for a class of linear dynamic systems with time- delay. In *Proceedings of the 2nd Asian Control Conference*, Seoul, Korea, July 1997.
- [182] M. Phan. A formulation of learning control for trajectory tracking using basis functions. In *Proceedings of the 35th IEEE Conference on Decision and Control*, Kobe, Japan, December 1996.
- [183] M. Phan and R. W. Longman. Liapunov based model reference learning control. In *Proceedings of the Twenty-Sixth Annual Allerton Conference on Communication, Control and Computing*, pages 927–936, University of Illinois at Urbana-Champaign, Coordinated Science Laboratory, September 1988.
- [184] M. Phan and R. W. Longman. A mathematical theory of learning control for linear discrete multivariable systems. In *Proceedings of the AIAA/AAS Astrodynamics Conference*, pages 740–746, Minneapolis, Minnesota, 1988.
- [185] M. Phan and R. W. Longman. Indirect learning control with guaranteed stability. In *Proceedings of the 1989 Conference on Information Sciences and Systems*, pages 125–131, Johns Hopkins University, Department of Electrical and Computer Engineering, Baltimore, MD, 1989.
- [186] M. Phan, Y. Wei, L. Horta, J. Juang, and R. Longman. Learning control for slewing of a flexible panel. In *Proceedings of the 1989 VPI&SU/AIAA Symposium on Dynamics and Control of Large Structures*, ?, May 1989.
- [187] M.Q. Phan and M. Chew. Application of learning control theory to mechanisms, part 1: Inverse kinematics and parametric error compensation. In *Proceedings of the 23rd ASME Mechanisms Conference*, Minneapolis, MN, September 1994.
- [188] M.Q. Phan and M. Chew. Application of learning control theory to mechanisms, part 2: reduction of residual vibrations in electromechanical bonding machines. In *Proceedings of the 23rd ASME Mechanisms Conference*, Minneapolis, MN, September 1994.
- [189] M.Q. Phan and M. Chew. Synthesis of four-bar function generators by an iterative learning control procedure. In *Proceedings of the 25th ASME Mechanisms Conference*, Irvine, California, August 1996.
- [190] M.Q. Phan and J. Juang. Designs of learning controllers based on an auto-regressive representation of a linear system. *Journal of Guidance, Control, and Dynamics*, 19(2):355–362, 1996.

- [191] M.Q. Phan and R. Longman. Learning system identification. In *Proceedings of the 20th Annual Pittsburgh Conference on Modeling and Simulation*, Pittsburgh, PA, May 1989.
- [192] M.Q. Phan and H. Rabitz. Learning control of quantum-mechanical systems by identification of effective input-output maps. *Chemical Physics*, 217(2):389–400, 1997.
- [193] J.B. Plant. *Some Iterative Solutions in Optimal Control*. M.I.T. Press, Cambridge, Massachusetts, 1968.
- [194] A. N. Poo, K. B. Lim, and Y. X. Ma. Application of discrete learning control to a robotic manipulator. *Robotics and Computer-Integrated Manufacturing*, 12(1):55–64, March 1996.
- [195] E. Rogers and D.H. Owens. Lyapunov stability theory and performance bounds for a class of 2D linear systems. *Multidimensional Systems and Signal Processing*, 7:179–194, 1996.
- [196] Cheol Lae Roh, Moon No Lee, and Myung Jin Chung. ILC for non-minimum phase systems. *International Journal of Systems Science*, 27(4):419–424, April 1996.
- [197] Nai run Yu and Bai wu Wan. Iterative learning control for the dynamics in steady state optimization of industrial processes. In *Proceedings of the 2nd Asian Control Conference*, Seoul, Korea, July 1997.
- [198] Y. S. Ryu, R. W. Longman, E. J. Solcz, and J. de Jong. Experiments in the use of repetitive control in belt drive systems. In *Proceedings of the Thirty-second Annual Allerton Conference on Communication, Control and Computing*, pages 324–334, Coordinated Sciences Laboratory, University of Illinois at Urbana-Champaign, 1994.
- [199] S. Saab. Discrete-time learning control algorithm for a class of non-linear systems. In *Proceedings of the 1995 American Control Conference*, Seattle, WA, June 1995.
- [200] S. Saab, W. Vogt, and M. Mickle. Learning control algorithms for tracking slowly varying trajectories. *IEEE Transactions on Systems, Man, and Cybernetics – Part B: Cybernetics*, 27(4):657–670, 1997.
- [201] Samer S. Saab. On the P-type learning control. *IEEE Transactions on Automatic Control*, 39(11), November 1994.
- [202] Samer S. Saab. A discrete-time learning control algorithm for a class of linear time-variant systems. *IEEE Transactions on Automatic Control*, 40(6), June 1995.

- [203] Nader Sadegh. Synthesis of a stable discrete-time repetitive controller for MIMO systems. *Journal of Dynamic Systems, Measurement, and Control*, 117:92–98, March 1995.
- [204] Nader Sadegh and Kennon Guglielmo. A new repetitive controller for mechanical manipulators. *Journal of Robotic Systems*, 8:507–529, August 1991.
- [205] Nader Sadegh, Roberto Horowitz, and Wei-Wen Kao. A unified approach to the design of adaptive and repetitive controllers for robotic manipulators. *Journal of Dynamic Systems, Measurement, and Control*, 112:618–629, December 1990.
- [206] G. Sardidis. *Self-Organizing Control of Stochastic Systems*. Marcel Dekker, New York, New York, 1977.
- [207] Zvi Shiller and Hai Chang. Trajectory preshaping for high-speed articulated systems. *Journal of Dynamic Systems, Measurement, and Control*, 117:304–310, September 1995.
- [208] L. G. Sison and E. K. P. Chong. No-reset iterative learning control. In *Proceedings of the 35th IEEE Conference on Decision and Control*, volume 3, pages 3062–3063, Kobe, Japan, December 1996.
- [209] C. Smith and M. Tomizuka. Shock rejection for repetitive control using a disturbance observer. In *Proceedings of the 35th IEEE Conference on Decision and Control*, Kobe, Japan, December 1996.
- [210] T. Sogo and N. Adachi. Convergence rates and robustness of iterative learning control. In *Proceedings of 35th IEEE Conference on Decision and Control*, volume 3, pages 3050–3055, Kobe, Japan, December 1996.
- [211] K. Srinivasan and F. R. Shaw. Analysis and design of repetitive control systems using the regeneration spectrum. *Journal of Dynamic Systems, Measurement, and Control*, 113:216–222, June 1991.
- [212] D. A. Streit, C. M. Krousgrill, and A. K. Bajaj. A preliminary investigation of the dynamic stability of flexible manipulators performing repetitive tasks. *Journal of Dynamic Systems, Measurement, and Control*, 108:206–214, September 1986.
- [213] D. A. Streit, C. M. Krousgrill, and A. K. Bajaj. Nonlinear response of flexible robotic manipulators performing repetitive tasks. *Journal of Dynamic Systems, Measurement, and Control*, 111:470–480, September 1989.

- [214] Y. Suganuma and M. Ito. Learning control and knowledge representation. In *Proceedings of IEEE International Symposium on Intelligent Control*, pages 354–359, Philadelphia, Pennsylvania, January 1987.
- [215] T. Sugie and T. Ono. An iterative learning control law for dynamical systems. *Automatica*, 27:729–732, 1991.
- [216] Dong Sun and Xiaolun Shi Yunhui Liu. Adaptive learning control for cooperation of two robots manipulating a rigid object with model uncertainties. *Robotica*, 14(4):365–373, July-August 1996.
- [217] M Sun, Y. Chen, and H. Dou. Robust higher order repetitive learning control algorithm for tracking control of delayed systems. In *Proceedings of the 1992 Conference on Decision and Control*, Tucson, AZ, December 1992.
- [218] M. Sun, B. Huang, X. Zhang, and Y. Chen. Robust convergence of D-type learning controller. In *Proceedings of the 2nd Chinese World Congress on Intelligent Control and Intelligent Automation*, Xi'an, China, 1997.
- [219] Mingxuan Sun, Baojian Huang, Xuezhi Zhang, Yangquan Chen, and Jian-Xin Xu. Selective learning with a forgetting factor for trajectory tracking of uncertain nonlinear systems. In *Proceedings of the 2nd Asian Control Conference*, Seoul, Korea, July 1997.
- [220] R. Sutton. Learning to predict by the methods of temporal differences. *Machine Learning*, 3:9–44, 1988.
- [221] S.Wang. Inversion of nonlinear dynamical systems. Technical Report, Dept. of Electrical and Computer Engineering, University of California, Davis, California, 1988.
- [222] M. Togai and O. Yamano. Analysis and design of an optimal learning control scheme for industrial robots: A discrete system approach. In *Proceedings of 24th Conference on Decision and Control*, pages 1399–1404, Ft. Lauderdale, Florida, December 1985.
- [223] Masayoshi Tomizuka, Tsu Chin Tsao, and Kok Kia Chew. Analysis and synthesis of discrete-time repetitive controllers. *Journal of Dynamic Systems, Measurement, and Control*, 111:353–358, September 1989.
- [224] S. K. Tso, Y. H. Fung, and N. L. Lin. Tracking improvement for robot manipulator control using neural-network learning. In *Proceedings of the 2nd Asian Control Conference*, Seoul, Korea, July 1997.

- [225] S. K. Tso and Y. X. Ma. Cartesian-based learning control for robots in discrete-time formulation. *IEEE Transactions on Systems, Man, and Cybernetics*, 22, September 1992.
- [226] Y.A. Tsyppkin. *Adaptation and Learning in Automatic Systems*. Academic Press, New York, New York, 1971.
- [227] Y.A. Tsyppkin. *Foundations of the Theory of Learning Systems*. Academic Press, New York, New York, 1973.
- [228] E. D. Tung, G. Anwar, and M. Tomizuka. Low velocity friction compensation and feedforward solution based on repetitive control. *Journal of Dynamic Systems, Measurement, and Control*, 115:279–284, June 1993.
- [229] M. Uchiyama. Formation of high speed motion pattern of mechanical arm by trial. *Transactions of the Society of Instrumentation and Control Engineers*, 19(5):706–712, May 1978.
- [230] Y. Uno, Y. Kawanishi, M. Kanda, and R. Suzuki. Iterative control for a multi-joint arm and learning of an inverse-dynamics model. *Systems and Computers in Japan*, 24(14):38–47, December 1993.
- [231] K. P. Venugopal, A. S. Pandya, and R. Sudhakar. A recurrent neural network controller and learning algorithm for the on-line learning control of autonomous underwater vehicles. *Neural Networks*, 7(5):833–846, 1994.
- [232] K. P. Venugopal, R. Sudhakar, and A. S. Pandya. On-line learning control of autonomous underwater vehicles using feedforward neural networks. *IEEE Journal of Oceanic Engineering*, 17:308–319, October 1992.
- [233] M. Vidyasagar. *A Theory of Learning and Generalization*. Springer-Verlag, New York, New York, 1997.
- [234] M.A. Waddoups and K.L. Moore. Neural networks for iterative learning control. In *Proceedings of 1992 American Control Conference*, Chicago, Illinois, June 1992.
- [235] S. Wang. Computed reference error adjustment technique (CREATE) for the control of robot manipulators. In *Proceedings of 22nd Annual Allerton Conference*, pages 874–876, October 1984.
- [236] S. Wang and I. Horowitz. CREATE – a new adaptive technique. In *Proceedings of 19th Annual Conference on Information Sciences and Systems*, March 1985.

- [237] Y. Wang and R. W. Longman. Use of non-causal digital signal processing in learning and repetitive control. *Advances in the Astronautical Sciences*, 90:649–668, 1996.
- [238] Chen Wen and Liu Deman. Learning control for a class of nonlinear systems with exogenous disturbances. *International Journal of Systems Science*, 27(12):1453–1459, December 1996.
- [239] J.-X. Xu. Direct learning of control input profiles with different time scales. In *Proceedings of the 35th IEEE Conference on Decision and Control*, Kobe, Japan, December 1996.
- [240] J.-X. Xu and Z. Qu. Robust learning control for a class of non-linear systems. In *Proceedings of the 35th IEEE Conference on Decision and Control*, Kobe, Japan, December 1996.
- [241] Jian-Xin Xu, Y. Dote, Xiaowei Wang, and Coming Shun. On the instability of iterative learning control due to sampling delay. In *Proceedings of the 1995 IEEE IECON 31st International Conference on Industrial Electronics, Control, and Instrumentation*, volume 1, pages 150–155, Orlando, FL, November 1995.
- [242] Jian-Xin Xu, Lee Tong Heng, and H. Nair. A revised iterative learning control strategy for robotic manipulators. In *Proceedings 1993 Asia-Pacific Workshop on Advances in Motion Control*, pages 88–93, Singapore, July 1993.
- [243] Jian-Xin Xu and Yanbin Song. Direct learning control of high-order systems for trajectories with different time scales. In *Proceedings of the 2nd Asian Control Conference*, Seoul, Korea, July 1997.
- [244] Jian-Xin Xu and Yanbin Song. Direct learning control scheme with an application to a robotic manipulator. In *Proceedings of the 2nd Asian Control Conference*, Seoul, Korea, July 1997.
- [245] Jian-Xin Xu, Xiao-Wei Wang, and Tong Heng Lee. Analysis of continuous iterative learning control systems using current cycle feedback. In *Proceedings of 1995 American Control Conference - ACC'95*, volume 6, pages 4221–4225, Seattle, WA, June 1995.
- [246] Jian-Xin Xu and Tao Zhu. Direct learning control of trajectory tracking with different time and magnitude scales for a class of nonlinear uncertain systems. In *Proceedings of the 2nd Asian Control Conference*, Seoul, Korea, July 1997.
- [247] W. Xu, C. Shao, and T. Chai. Learning control for a class of constrained mechanical systems with uncertain disturbances. In *Proceedings of the 1997 American Control Conference*, Albuquerque, NM, June 1997.

- [248] M. Yamakita and K. Furuta. Iterative generation of virtual reference for a manipulator. *Robotica*, 9:71–80, 1991.
- [249] Tae yong Kuc, Kwanghee Nam, and Jin S. Lee. An iterative learning control of robot manipulators. *IEEE Transactions on Robotics and Automation*, 7:835–842, December 1991.
- [250] K. Y. Yu, C. H. Choi, and T. J. Jang. An iterative learning control for the precision improvement of a CNC machining center. In *Proceedings of the 5th International Conference on Signal Processing Applications and Technology*, volume 2, pages 1055–1060, Dallas, TX, October 1994.
- [251] B. S. Zhang and J. R. Leigh. Predictive time-sequence iterative learning control with application to a fermentation process. In *Proceedings of IEEE International Conference on Control and Applications*, volume 2, pages xviii, viii, 982, Vancouver, BC, Canada, September 1993.
- [252] Jian-Hua Zhang and Li-Min Jia. Coordination level in hierarchical intelligent control systems. In *Proceedings of the 2nd Asian Control Conference*, Seoul, Korea, July 1997.
- [253] Zhaowei Zhong. An application of H-infinity control and iterative learning control to a scanner driving system. In *Proceedings of 3rd International Conference on Computer Integrated Manufacturing*, volume 2, pages 981–988, Singapore, July 1995.
- [254] A. Zilouchian. An iterative learning control technique for a dual arm robotic system. In *Proceedings of the 1994 IEEE International Conference on Robotics and Automation*, volume 4, pages 1528–1533, San Diego, CA, May 1994.