Chapter 1
Introduction

This chapter provides a rudimentary introduction of iterative learning control (ILC) and its basic formulation for both discrete-time and continuous-time systems, which is followed by a review on recent developments of ILC with iteration-varying trial lengths. At the end of this chapter, the structure/organization of the whole monograph is also presented.

1.1 Iterative Learning Control

In our daily life, almost every task is conducted by trial and error, which is the inherent concept of learning. Indeed, it is learning that helps us to survive from severe conditions in the ancient times and become stronger in handling with various problems nowadays. The underlying philosophy of the human being learning process is "practice makes perfect". Human being can do things better and better after several practices. For example, when we learn to write a Chinese character in primary school, our teacher always asks us to repeat the character many times. While repeating the character, we are adjusting the writing positions step by step and the writing performance will be improved gradually. Another example is the basketball shooting. Consider to learn to shoot the basketball from a fixed position, it might be difficult for us to hit the basket at the first several trials since we have little knowledge about the correct shooting angle and force. The hit ratio will be definitely increased if we can learn from the failures and correct our behavior.

Learning is an important concept to human being, and it is interesting to find that such a fundamental principle can be applied to control systems, which is the origin of iterative learning control (ILC). ILC is designed for systems repeating a certain task over a fixed time interval. In ILC, the current control input signal is generated by previous input and output information as well as the tracking objective. As a result, the control performance can be gradually improved as the iteration number increases.

D. Shen and X. Li, *Iterative Learning Control for Systems with Iteration-Varying Trial Lengths*, https://doi.org/10.1007/978-981-13-6136-4_1

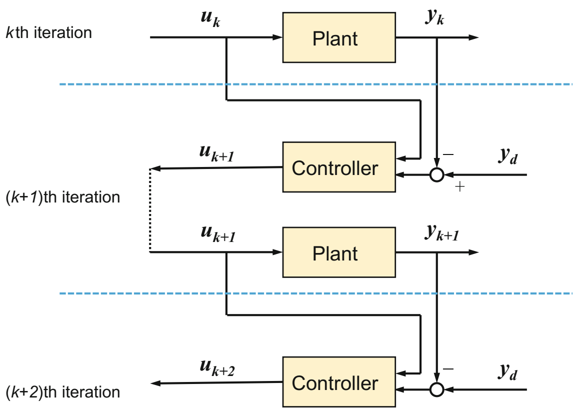

Fig. 1.1 ILC in iteration domain

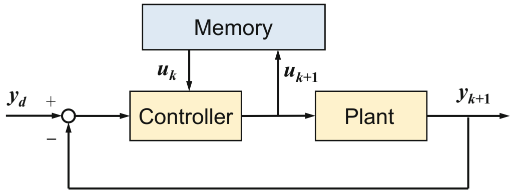

Fig. 1.2 Framework of ILC

Note that the term “iteration” may be replaced with “trial”, “batch”, and “cycle” in the literature due to different backgrounds.

In contrast to other control methods, such as adaptive control and robust control, ILC has several distinct features, which include (1) finite time horizon, (2) accurate resetting condition, and (3) perfect repeating conditions including the system plant and control objective.

The block diagram of ILC is shown in Fig. 1.1, where y_d denotes the reference trajectory. At the kth iteration, the input u_k is fed to the plant so that the corresponding output is y_k and the tracking error is $e_k = y_d - y_k$. This is a causal process. Since the tracking error e_k is nonzero, implying that the input u_k is not good enough, the input signal of the next iteration (i.e., the $(k+1)$th iteration) should then be updated. The control input signal u_{k+1} at the $(k+1)$th iteration usually is constructed as a function of u_k and e_k. In other words, the control input signal u_{k+1} is generated by previous control information. Then, the newly generated input is fed to the plant for the next iteration. Meanwhile, the input u_{k+1} is also stored into the memory for updating the input for the $(k+2)$th iteration. As a result, a closed-loop feedback is formed in the iteration domain. A concise block diagram of the ILC principle is shown in Fig. 1.2, which is common in many ILC papers and monographs.

As can be seen from the above figures, the main difference between ILC and the conventional control methodologies is that ILC improves the control performance along the iteration axis rather than the time axis. In other words, the control performance is gradually improved as the iteration number increases to infinity, while the transient performance within an iteration is usually ignored. Meanwhile, ILC can be combined with the conventional control methods to further enhance performance. That is, we can employ the conventional control method as an inner control loop to achieve an acceptable performance of the systems such as stability and then add ILC as an outer control loop to further improve the tracking precision. This topic has also been investigated in the literature.

As discussed before, ILC synthesizes the current control input signal from previous input and output information, while the exact system plant information is not required. Actually, this is one of the major advantages of ILC. In other words, ILC is a data-driven control method.

ILC has been proposed for a long time which can be tracked back to a US patent [1] in 1967. In 1978, Uchiyama initialized a “repeating” method of correcting the reference function by trial [2]. However, this paper was written in Japanese and failed to attract wide attention in the community. The paper published by Arimoto et al. in 1984 opened the research of ILC [3]. Since then, numerous articles were published on ILC. Detailed surveys on ILC can be found in [4–9].

1.2 Basic Formulation of ILC

Now, let us go through the basic formulation of ILC for both discrete-time and continuous-time systems, which is followed by the conventional convergence analysis.

1.2.1 Discrete-Time Case

Consider the following discrete time-invariant linear system

$$\begin{aligned} x_k(t+1) &= Ax_k(t) + Bu_k(t), \\ y_k(t) &= Cx_k(t), \end{aligned} \tag{1.1}$$

where $x \in \mathbf{R}^n$, $u \in \mathbf{R}^p$, and $y \in \mathbf{R}^q$ denote the system state, input, and output, respectively. Matrices A, B, and C are system matrices with appropriate dimensions. t denotes an arbitrary time instant in an operation iteration, $t = 0, 1, \ldots, N$, where N is the length of the operation iteration. For simplicity, $t \in [0, N]$ is used in the following. $k = 0, 1, 2, \ldots$ denote different iterations.

Since it is required that a given tracking task should be repeated, the initial state needs to be reset at each iteration. The following is a basic reset condition, called identical initialization condition (i.i.c.), which is common in ILC theory.

$$x_k(0) = x_0, \quad \forall k. \tag{1.2}$$

The reference trajectory is denoted by $y_d(t)$, $t \in [0, N]$. With regard to the reset condition, it is usually required that $y_d(0) = y_0 \triangleq Cx_0$. The control objective of ILC is to design a proper update law for the input $u_k(t)$, so that the corresponding output $y_k(t)$ can track $y_d(t)$ as closely as possible. To this end, for any t in $[0, N]$, we define the tracking error as

$$e_k(t) = y_d(t) - y_k(t). \tag{1.3}$$

Then, the update law is a function of $u_k(t)$ and $e_k(t)$ to generate $u_{k+1}(t)$, whose general form is as follows:

$$u_{k+1}(t) = h(u_k(\cdot), \ldots, u_0(\cdot), e_k(\cdot), \ldots, e_0(\cdot)). \tag{1.4}$$

When the above relationship depends only on the last iteration, it is called first-order ILC update law; otherwise, it is called high-order ILC update law. Generally, to achieve the algorithm simplicity, most update laws are of first order, i.e.,

$$u_{k+1}(t) = h(u_k(\cdot), e_k(\cdot)). \tag{1.5}$$

Additionally, the update law is usually linear. The simplest update law is as follows:

$$u_{k+1}(t) = u_k(t) + Ke_k(t+1), \tag{1.6}$$

where K is the learning gain matrix to be designed. In (1.6), $u_k(t)$ is the input of current iteration, while $Ke_k(t+1)$ is the innovation term. The update law (1.6) is called P-type ILC update law. If the innovation term is replaced by $K(e_k(t+1) - e_k(t))$, the update law is called D-type.

For system (1.1) and update law (1.6), a basic convergence condition is that K satisfies

$$\|I - CBK\| < 1. \tag{1.7}$$

Then, one has $\|e_k(t)\| \xrightarrow[k\to\infty]{} 0$, where $\|\cdot\|$ denotes the matrix or vector norm.

From this condition, one can deduce that the design of K needs no information with regard to the system matrix A, but requires information of the coupling matrix CB. This fact demonstrates the advantage of ILC from the perspective that ILC has little dependence on the system information A. Thus, ILC can handle tracking problems that have more uncertainties.

Remark 1.1 From the formulation of ILC, one can see that the model takes the classic features of a 2D system. That is, the system dynamics (1.1) and the up date law (1.6) evolve along time and iteration axes, respectively. Many scholars have made contributions from this point of view and developed a 2D system-based approach, which is one of the principal techniques for ILC design and analysis.

Note that the operation length is limited by N, and is then repeated multiple times. Thus, one could use the so-called lifting technique for discrete-time systems, which implies lifting all of the inputs and outputs as supervectors,

$$U_k = [u_k^T(0), u_k^T(1), \ldots, u_k^T(N-1)]^T, \tag{1.8}$$
$$Y_k = [y_k^T(1), y_k^T(2), \ldots, y_k^T(N)]^T. \tag{1.9}$$

Denote

$$G = \begin{bmatrix} CB & 0 & 0 & \cdots & 0 \\ CAB & CB & 0 & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ CA^{N-1}B & CA^{N-2}B & \cdots & \cdots & CB \end{bmatrix}, \tag{1.10}$$

then, we have

$$Y_k = GU_k + \mathbf{d}, \tag{1.11}$$

where

$$\mathbf{d} = [(Cx_0)^T, (CAx_0)^T, \ldots, (CA^{N-1}x_0)^T]^T. \tag{1.12}$$

Similar to (1.8) and (1.9), define

$$Y_d = [y_d^T(1), y_d^T(2), \ldots, y_d^T(N)]^T,$$
$$E_k = (e_k^T(1), e_k^T(2), \ldots, e_k^T(N))^T,$$

then it leads to

$$U_{k+1} = U_k + \mathbf{K}E_k, \tag{1.13}$$

where $\mathbf{K} = \text{diag}\{K, K, \ldots, K\}$. By simple calculation, we have

$$\begin{aligned} E_{k+1} &= Y_d - Y_{k+1} = Y_d - GU_{k+1} - \mathbf{d} \\ &= Y_d - GU_k - G\mathbf{K}E_k - \mathbf{d} \\ &= E_k - G\mathbf{K}E_k \\ &= (I - G\mathbf{K})E_k. \end{aligned}$$

Therefore, we obtain the condition (1.7) that is sufficient to guarantee the convergence of ILC. Actually, the lifting technique does not only help us to obtain the convergence condition but also provides us with an intrinsic understanding of ILC. In the lifted model (1.11), the time-domain evolutionary process within the operating iteration has been integrated into G, whereas the relationship between adjacent iterations is highlighted. That is, the lifted model (1.11) depends on the iteration axis only.

Remark 1.2 Note that the focus of ILC is how to improve the tracking performance gradually along the iteration axis, as one can see from the design of the update law

(1.13) and lifted model (1.11). Therefore, it would not cause additional difficulties when the system is extended from the linear time-invariant case to the linear time-varying case. This is because, for any fixed time, the updating process along the iteration axis is a time-invariant case.

It is usually assumed that the reference trajectory $y_d(t)$ is realizable. That is, there exists an appropriate initial state x_0 and input $u_d(t)$ such that the expression (1.1) still holds with the subscript k being replaced by d. In other words, $Y_d = GU_d + \mathbf{d}$, where U_d is defined in a similar manner as (1.8). Then, the discussion that the system output converges to the reference trajectory, i.e., $\lim_{k\to\infty} Y_k = Y_d$, is equivalent to the one that the system input converges to the objective input, i.e., $\lim_{k\to\infty} U_k = U_d$. For the system with stochastic noises, this transformation of proof objective is more convenient for convergence analysis.

Remark 1.3 One may be interested in the case that the reference trajectory is not realizable. In other words, there is no control input producing the reference trajectory; thus, entirely accurate tracking is impossible. Then, the design objective of the ILC algorithm is no longer to guarantee asymptotically accurate tracking, but to converge to the nearest trajectory of the given reference. Consequently, the tracking problem has become an optimization problem. On the other hand, from the viewpoint of practical applications, the reference trajectory is usually realizable; thus, the assumption is not rigorous.

1.2.2 Continuous-Time Case

Let us consider the following linear continuous-time system:

$$\begin{aligned} \dot{x}_k(t) &= Ax_k(t) + Bu_k(t), \\ y_k(t) &= Cx_k(t), \end{aligned} \tag{1.14}$$

where notations have similar meanings to the discrete-time case.

The control task is to drive the output y_k to track the desired reference y_d on a fixed time interval $t \in [0, T]$ as the iteration number k increases. If the relative degree of the system is one, an ILC scheme of Arimoto type can be given as

$$u_{k+1} = u_k + \varGamma \dot{e}_k, \tag{1.15}$$

where $e_k(t) = y_d(t) - y_k(t)$ and $\varGamma$ is the diagonal learning gain matrix. Similarly, if the learning gain matrix satisfies

$$\|I - CB\varGamma\| < 1, \tag{1.16}$$

then the control objective can be achieved, i.e., $\lim_{k\to\infty} y_k(t) \to y_d(t)$. Note that the basic formula for selecting the learning gain matrix given in (1.16) requires no information about the system matrix A, which implies that ILC is effective for uncertain system matrices.

Moreover, a "PID-like" update law can be formulated as

$$u_{k+1} = u_k + \Phi e_k + \Gamma \dot{e}_k + \Psi \int e_k \mathrm{d}t, \tag{1.17}$$

where Φ, Γ, and Ψ are learning gain matrices. The high-order PID-like update law can be formulated as

$$u_{k+1} = \sum_{i=1}^{n} (I - \Lambda) P_i u_{k+1-i} + \Lambda u_0 + \sum_{i=1}^{n} \left(\Phi_i e_{k+1-i} + \Gamma_i \dot{e}_{k+1-i} + \Psi_i \int e_{k+1-i} \mathrm{d}t \right), \tag{1.18}$$

where $\sum_{i=1}^{n} P_i = I$.

1.3 ILC for Systems with Varying Trial Lengths

In traditional ILC, to achieve perfect tracking performance, some exactly repeating conditions, such as identical trial length, identical initial condition, and iteration-invariant learning target, are required. However, these iteration-invariant conditions will often be violated in real-time applications due to unknown uncertainties or unpredictable factors, which hinders practical applications of conventional ILC, and thus motivate scholars to relax/remove the perfect repeating conditions in ILC. This monograph will focus on ILC design when control systems have iteration-varying trial lengths. In practice, sometimes it is difficult to ensure that the control system repeats on a fixed time interval. For instance, when applying ILC in a functional electrical stimulation (FES) for upper limb movement and gait assistance, it is found that the operation processes end early for at least the first few passes due to safety considerations [10]. The FES-induced foot motion and the associated variable-length-trial problem are detailed in [11, 12], which clearly illustrate the violation of the identical-trial-length assumption. Another example can be seen in the analysis of humanoid and biped walking robots, which is characterized by periodic or quasi-periodic gaits [13]. For analysis purpose, these gaits are divided into phases that are defined by the time when the foot strikes the ground, and the duration of the resulting phases are usually not the same from iteration to iteration. Furthermore, as can be found in [14], a trajectory-tracking problem of a lab-scale gantry crane was investigated under the framework of ILC. In this example, the trial lengths at different iterations might be varying since authors defined that the learning process should be terminated if the system output drift far away from the desired trajectory. Based on these observations, it is interesting and valuable to investigate ILC with iteration-varying trial lengths.

To clarify the effect of varying trial lengths, one can refer to Fig. 1.3, where Fig. 1.3a illustrates the complete trial length with T_d being the desired iteration length while Fig. 1.3b–d demonstrates possible incomplete trial lengths. In other words, the varying trial length problem here indicates that the iteration may end before its desired

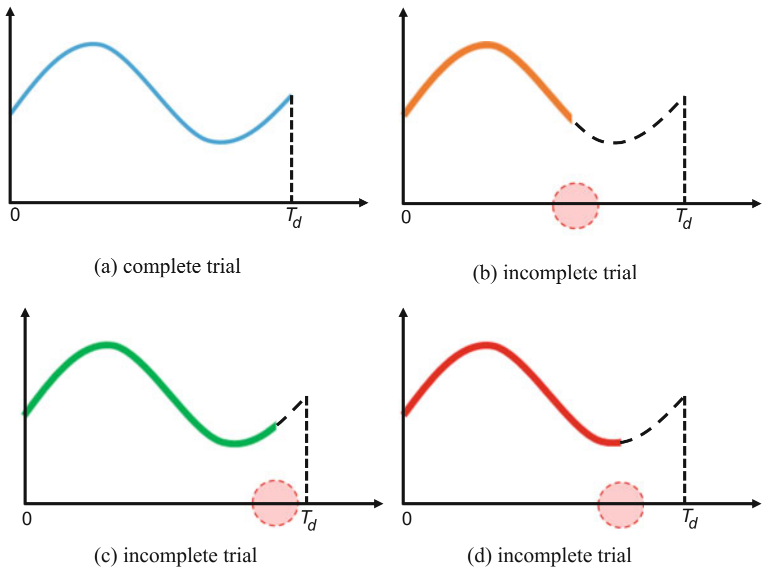

Fig. 1.3 Illustration of varying trial lengths

time length but the tracking objective remains the same for all iterations. Therefore, the major influence of this setting is that the latter part of the tracking information is lost if the iteration ends early. As a result, emphasis of analysis should go to the inherent effect of uncompleted operation part. Moreover, in this monograph, we consider the varying length to be random with or without statistical knowledge, and thus the specific convergence sense should be carefully considered.

There were some early research aiming to provide a suitable design and analysis framework for varying-iteration-length ILC that formed the groundwork for subsequent investigations [10–14]. For example, based on the experimental verifications and primary analysis of the convergence property given in [10–12], a systematic proof of the monotonic convergence in different norm senses was elaborated in [15] for linear systems with nonuniform trial lengths. In that paper, the necessary and sufficient conditions for monotonic convergence were discussed, as well as other issues including the controller design guidelines and influence of disturbances. However, it is worthwhile to mention that authors did not provide an uniform framework on ILC with iteration-varying trial lengths.

The first random model of varying-length iterations was proposed in [16] for discrete-time systems, and it was then extended to continuous-time systems in [17]. In [16] and [17], a stochastic variable was used to represent the occurrence of the output at each time instant and iteration, and it was then multiplied to the tracking error, which denoted the actual information of the updating process. To compensate for the information loss caused by randomly varying trial lengths, an iteration-average operator of all historical data was introduced to the ILC algorithm in [16], whereas in [17], this average operator was replaced by an iteration-moving-average operator to reduce the influence of very old data. Moreover, a lifted framework of ILC for a discrete-time linear system was provided in [18] to avoid conservatism of the conventional λ-norm-based analysis in [16, 17].

Note that all of the results in [16–18] obtained asymptotical convergence with respect to the expected value, which is very weak for the control of stochastic models, and thus motivates scholars to seek a stronger convergence result in [19]. In detail, the discrete-time linear system was revisited and the traditional P-type ILC law was employed. The authors transformed the error evolution along the iteration axis by modeling it as a switching system and then established the input error's statistical properties (i.e., the mathematical expectations and covariances) in a recursive form. The convergence in the mathematical expectation, mean square, and almost sure senses was derived simultaneously. The results were then extended to a class of affine nonlinear systems in [20] using different analysis techniques. A recent work [21] further proposed two novel and improved ILC schemes based on the iteration-moving-average operator, in which a random searching mechanism was additionally introduced to collect useful information while avoiding redundant tracking information from the past.

In addition, some extensions have also been reported. Nonlinear stochastic systems were taken into account in [22] with bounded disturbances. Nevertheless, a Gaussian distribution of the variable pass length was required, which limits the possible application range. In [23], the authors extended the method to discrete-time linear systems with a vector relative degree. In this case, one needs to carefully select the output data for the learning algorithms to function. The issue was also extended to stochastic impulse differential equations in [24] and fractional order systems in [25]. We would like to note that the convergence analyses derived in these papers were primarily based on the mature contraction mapping method similar to [16]. A recent progress [26] presented a deterministic convergence analysis if the full-length learning occur any adjacent finite iterations.

In short, we can observe the following facts from the above literature. First, most papers have focused on discrete-time linear systems such as those in [15, 16, 18, 19, 21, 23], mainly owing to the beneficial system structure and mature analysis techniques for discrete random variables. The results on continuous-time systems, originated from practical systems, are rather limited. Although nonlinear systems were considered in [17, 20, 22], the globally Lipschitz continuous condition was imposed on the nonlinear functions in these papers, which effectively transforms the system to a linear system. Therefore, it is significant to consider removing the globally Lipschitz continuous condition for continuous-time systems.

This advance was presented in a recent paper [27], where continuous-time parameterized nonlinear systems with nonlinear functions being locally Lipschitz continuous were taken into account. Adaptive learning controller consisting of a stabilization feedback term and a compensation feedforward term was proposed. A novel modified composite energy function (CEF) was defined with the new concept of a virtual tracking error for the untrodden part of each iteration. This CEF allowed one to present an explicit difference between adjacent iterations and thus facilitated the analysis. Moreover, if partial structure information is available, the paper [28] presented two types of learning schemes: a mixing-type adaptive learning scheme and a hybrid-type differential-difference learning scheme. The convergence was conducted using similar idea of [27] with novel virtual tracking errors.

1.4 Structure of this Monograph

In this monograph, we concentrate on ILC for systems with varying trial lengths. Our primary objective is to provide a systematic framework of the synthesis and analysis of ILC algorithms. To this end, we will clarify the following aspects: the controller design, the convergence analysis, and the influence evaluation of nonuniform trial lengths. The investigation of this monograph would greatly help to understand ILC with varying trial lengths, which is a specific type of incomplete information.

The visual structure of this monograph is shown in Fig. 1.4, where three different divisions of the chapters can be observed.

First division: The main materials in this monograph are placed into two parts. Part I including Chaps. 2–6 focuses on linear systems, for which the conventional P-type algorithm can behave well. Part II including Chaps. 7–12 aims to provide fruitful results for nonlinear systems, where direct and indirect learning schemes are proposed.

Second division: Readers can also refer to the monograph according to discrete-time and continuous-time types of system dynamics. In particular, Chaps. 2–6 and 8 present the design and analysis techniques for discrete-time systems and Chaps. 7, 8–12 are the counterpart for continuous-time systems. It should be mentioned that the analysis for discrete time and continuous time are fairly different.

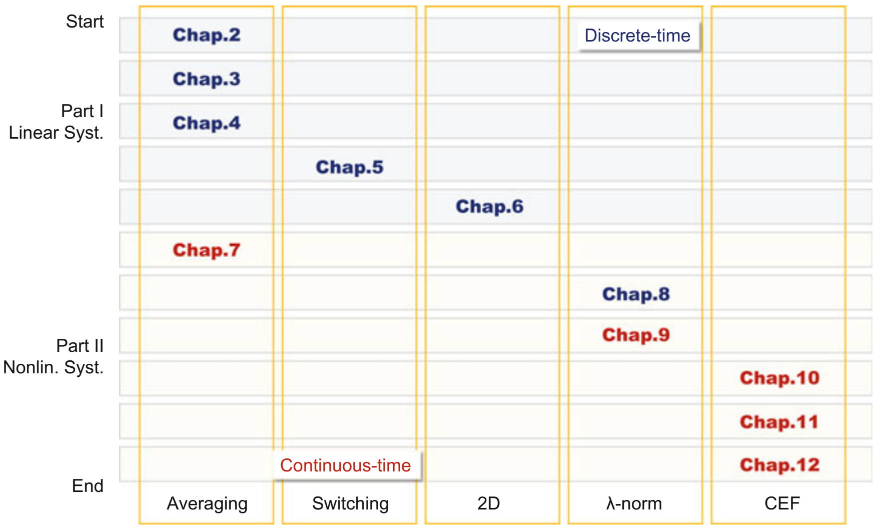

Fig. 1.4 Structure of this monograph

Third division: The monograph also provides a technical angle of the results. Specifically, Chaps. 2–4 and 7 deeply discuss the averaging technique in controller design. Chapter 5 gives a novel switching system approach for convergence analysis of the conventional P-type scheme. Chapter 6 presents the two-dimensional Kalman-filtering-based approach for stochastic systems. Chapters 8 and 9 provide an in-depth application of the λ-norm technique for analysis. Chapters 10–12 comprehensively show the definition and employment of CEF techniques for continuous-time nonlinear systems.

The contents of each chapter can be summarized as follows:

Chapter 2 presents a novel formulation and idea to address the tracking control problem for discrete-time linear systems with randomly varying trial lengths. An ILC scheme with an iteration-average operator is introduced, which thus mitigates the requirement on classic ILC that all trial lengths must be identical.

Chapter 3 also considers a class of discrete-time linear systems with randomly varying trial lengths. However, in contrast to Chap. 2, this chapter aims to avoid using the traditional λ-norm in convergence analysis which may lead to a non-monotonic convergence.

Chapter 4 proposes two novel ILC schemes for discrete-time linear systems with randomly varying trial lengths. In contrast to Chaps. 2 and 3 that advocate to replace the missing control information by zero, the proposed learning algorithms in this chapter are equipped with a random searching mechanism to collect useful but avoid redundant past tracking information, which could expedite the learning speed.

Chapter 5 proceeds to a novel analysis technique for linear discrete-time systems, which is called the switching system technique. In this technique, the iteration evolution of the input error is formulated as a switching system. Then, the mean and covariance of the associated random matrices can be recursively computed along the iteration axis, which paves a novel way for convergence analysis.

Chapter 6 presents the two-dimensional technique for addressing the tracking problem of linear discrete-time stochastic systems with varying trial lengths. The Kalman filtering technique is applied to derive the recursive learning gain matrix which guarantees the mean square convergence of the input error to zero. As a consequence, the tracking error will converge asymptotically in mean square sense.

Chapter 7 extends the idea on ILC design with randomly varying trial lengths to nonlinear continuous-time dynamic systems. Different from Chaps. 2 and 3, this chapter will employ an iteratively moving average operator with fixed window length into the ILC scheme.

Chapter 8 considers the discrete-time nonlinear systems, which is different from the continuous-time case in the previous chapter. In particular, the affine nonlinear system is taken into account, where the nonlinear functions satisfy globally Lipschitz continuous condition. A novel technical lemma is also provided for the strict convergence analysis in pointwise sense.

Chapter 9 provides the first result on sampled-data control for continuous-time nonlinear systems with varying trial lengths. To deal with the iteration-varying length problem, we propose two sampled-data ILC schemes, a generic PD-type scheme and a modified version with moving average operator, based on the modified tracking errors that have been redefined when the trial length is shorter or longer than the desired one. Sufficient conditions are derived rigorously to guarantee the convergence of the nonlinear system at each sampling instant.

Chapter 10 proposes a novel method for parameterized nonlinear continuous-time systems with varying trial lengths. As opposed to the previous chapters, this chapter is applicable to nonlinear systems that do not satisfy the globally Lipschitz continuous condition. To solve the problem, the adaptive ILC schemes are adopted in this chapter to learn the parameters and ensure an asymptotical convergence. Moreover, this chapter introduces a novel CEF using newly defined virtual tracking errors for proving the convergence.

Chapter 11 proceeds to consider the continuous-time nonlinear systems with non-parametric uncertainties, differing the parameterized systems in the previous chapter, under nonuniform trial length circumstances. Three common types of nonparametric uncertainties are taken into account in sequence: norm-bounded uncertainty, variation-norm-bounded uncertainty, and norm-bounded uncertainty with unknown coefficients. The CEF defined in the previous chapter is employed for the asymptotical convergence of the proposed schemes.

Chapter 12 applies the CEF technique proposed in Chaps. 10 and 11 to uncertain systems with two specific types of partial structure information. First, we consider the case that the system uncertainty consists of two parts, a time-invariant part and a time-varying part. A mixing-type adaptive learning scheme is derived, where the time-invariant part and the time-varying part are learned in differential and difference forms. Next, we move to consider the case that time-invariant and time-varying system uncertainties cannot be directly separated. A hybrid form of the differential and difference learning laws is proposed, where both differential and difference learning mechanisms are integrated in a unified adaptive learning scheme to derive the estimation of unknown parameters.

1.5 Summary

In this chapter, the introduction of ILC is provided first, which is followed by the basic formulation of ILC for both discrete-time and continuous-time control systems. In addition, a brief review of ILC with iteration-varying trial lengths is then given. Lastly, the structure of the whole monograph is also presented.

References

1. Garden M (1967) Learning control of actuators in control systems. US Patent, Appl No 637769
2. Uchiyama M (1978) Formulation of high-speed motion pattern of a mechanical arm by trial. Trans Soc Instrum Control Eng 14(6):706–712
3. Arimoto S, Kawamura S, Miyazaki F (1984) Bettering operation of robots by learning. J Robotic Syst 1(2):123–140
4. Bristow DA, Tharayil M, Alleyne AG (2006) A survey of iterative learning control: a learning-based method for high-performance tracking control. IEEE Control Syst Mag 26(3):96–114
5. Ahn H-S, Chen YQ, Moore KL (2007) Iterative learning control: survey and categorization from 1998 to 2004. IEEE Trans Syst Man Cybern Part C 37(6):1099–1121
6. Wang Y, Gao F, Doyle FJ III (2009) Survey on iterative learning control, repetitive control and run-to-run control. J Process Control 19(10):1589–1600
7. Shen D, Wang Y (2014) Survey on stochastic iterative learning control. J Process Control 24(12):64–77
8. Shen D (2018) Iterative learning control with incomplete information: a survey. IEEE/CAA J Autom Sin 5(5):885–901
9. Shen D (2018) A technical overview of recent progresses on stochastic iterative learning control. Unmanned Syst 6(3):147–164
10. Seel T, Schauer T, Raisch J (2011) Iterative learning control for variable pass length systems. In: Proceedings of the 18th IFAC World congress. Milano, Italy, pp 4880–4885, 28 Aug–2 Sept 2011
11. Seel T, Werner C, Schauer T (2016) The adaptive drop foot stimulator - multivariable learning control of foot pitch and roll motion in paretic gait. Med Eng Phys 38(11):1205–1213
12. Seel T, Werner C, Raisch J, Schauer T (2016) Iterative learning control of a drop foot neuroprosthesis - generating physiological foot motion in paretic gait by automatic feedback control. Control Eng Pract 48:87–97
13. Longman RW, Mombaur KD (2014) Investigating the use of iterative learning control and repetitive control to implement periodic gaits. Lect Notes Control Inf Sci 340:189–218
14. Guth M, Seel T, Raisch J (2013) Iterative learning control with variable pass length applied to trajectory tracking on a crane with output constraints. In: Proceedings of the 52nd IEEE conference on decision and control. Florence, Italy, pp 6676–6681
15. Seel T, Schauer T, Raisch J (2017) Monotonic convergence of iterative learning control systems with variable pass length. Int J Control 90(3):393–406
16. Li X, Xu J-X, Huang D (2014) An iterative learning control approach for linear systems with randomly varying trial lengths. IEEE Trans Autom Control 59(7):1954–1960
17. Li X, Xu J-X, Huang D (2015) Iterative learning control for nonlinear dynamic systems with randomly varying trial lengths. Int J Adapt Control Signal Process 29(11):1341–1353
18. Li X, Xu J-X (2015) Lifted system framework for learning control with different trial lengths. Int J Autom Comput 12(3):273–280
19. Shen D, Zhang W, Wang Y, Chien C-J (2016) On almost sure and mean square convergence of P-type ILC under randomly varying iteration lengths. Automatica 63:359–365
20. Shen D, Zhang W, Xu J-X (2016) Iterative learning control for discrete nonlinear systems with randomly iteration varying lengths. Syst Control Lett 96:81–87
21. Li X, Shen D (2017) Two novel iterative learning control schemes for systems with randomly varying trial lengths. Syst Control Lett 107:9–16
22. Shi J, He X, Zhou D (2016) Iterative learning control for nonlinear stochastic systems with variable pass length. J Frankl Inst 353:4016–4038
23. Wei Y-S, Li X-D (2017) Varying trail lengths-based iterative learning control for linear discrete-time systems with vector relative degree. Int J Syst Sci 48(10):2146–2156
24. Liu S, Debbouche A, Wang J (2017) On the iterative learning control for stochastic impulsive differential equations with randomly varying trial lengths. J Comput Appl Math 312:47–57
25. Liu S, Wang J (2017) Fractional order iterative learning control with randomly varying trial lengths. J Frankl Inst 354:967–992

26. Meng D, Zhang J (2018) Deterministic convergence for learning control systems over iteration-dependent tracking intervals. IEEE Trans Neural Netw Learn Syst 29(8):3885–3892
27. Shen D, Xu J-X (2018) Adaptive learning control for nonlinear systems with randomly varying iteration lengths. IEEE Trans Neural Netw Learn Syst. https://doi.org/10.1109/TNNLS.2018.2861216
28. Zeng C, Shen D, Wang J (2018) Adaptive learning tracking for uncertain systems with partial structure information and varying trial lengths. J Frankl Inst 355(15):7027–7055