DEVELOPMENT OF NEW LEARNING CONTROL APPROACHES

YAN RUI

NATIONAL UNIVERSITY OF SINGAPORE

2005

Founded 1905

DEVELOPMENT OF NEW LEARNING CONTROL APPROACHES

BY

YAN RUI

(M.Sci. Sichuan Univ.)

A THESIS SUBMITTED

FOR THE DEGREE OF DOCTOR OF PHILOSOPHY

DEPARTMENT OF ELECTRICAL AND COMPUTER ENGINEERING

NATIONAL UNIVERSITY OF SINGAPORE

2005

Acknowledgments

First and foremost, I would like to express my sincere thanks to my supervisor A/P. Jian-Xin Xu for his inspiration, valuable guidance, supports and encouragement throughout my research progress. His impressive academic achievements in the research areas of Learning Control and nonlinear Control attracted me to do the research work in Learning Control. Without his expert guidance and help, this thesis would never have come out. His endless enthusiasm, rigorous scientific approach and encouragement gave a strong impetus to my scientific work. Working with him proves to be a rewarding and pleasurable experience.

I would also like to thank Prof. Ben M. Chen and Dr. Cheng Xiang at National University of Singapore who provided me kind encouragement and constructive suggestions for my research. I also want to express my deep gratitude to Prof. Zhang Weinian (Sichun Univ.). I benefited much from influential discussions with him on the research topic. I am also grateful to all my laboratory-mates in the Control & Simulation Lab which provides good research facilities. I highly appreciate the friendly atmosphere and all the nice time we spent together in the last four years.

Special thanks go to my husband Tang Huajin, for his support, encouragement and love. Particularly, he gave me a lot of constructive suggestions to this thesis.

Finally, I dedicate this work to my parents for their supports and love through my life.

Contents

Summary

Learning control mainly aims at improving the system performance via directly updating the control input, either repeatedly over a fixed finite time interval, or repetitively (cyclically) over an infinite time interval. Moreover, there are two kinds of non-repeatable problems encountered in learning control: non-repeatability of a motion task and non-repeatability of a process. In this thesis, the attention is concentrated on the direct learning control (DLC), iterative learning control (ILC) and repetitive learning control (RLC) analysis and design. The main contributions of this thesis are to develop new learning control approaches for linear and nonlinear dynamic systems.

In the first part of the thesis, a DLC approach for a class of switched systems is proposed. The objective of direct learning is to generate the desired control profile for a newly switched system without any feedback, even if the system may have uncertainties. The DLC approach is achieved by exploring the inherent relationship between any two systems before and after a switch. The new approach is applicable to a class of linear time varying, uncertain, and switched systems, when the trajectory tracking control problem is concerned. Furthermore, singularity problem and trajectory switch problem are also considered.

In the second part of the thesis, four different ILC approaches are proposed.

(1). Two kinds of ILC approaches are presented by adding a forgetting factor and adopting a time varying learning gain to deal with input singularities problem. The proposed ILC approaches ensure a convergent control input sequence approaching to a unique fixed point based on Banach fixed point theorem. In the presence of the first type of singularities, the fixed point guarantees that the system output enters and remains uniformly in a designated neighborhood of the target trajectory. While in the presence of the second type of singularities, the tracking error is bounded by

a class $\mathcal{K}$ function of the designated neighborhood.

(2). To deal with the tracking problem without *a priori* knowledge of the control direction, an ILC approach is constructed with both differential and difference updating laws by incorporating a Nussbaum-type function. The new ILC approach can warrant a L_T^2 convergence of the tracking error sequence along the iteration axis, in the presence of time-varying parametric uncertainties and local Lipschitz nonlinearities.

(3). A new ILC approach is proposed to handle finite interval tracking problems based on constructive function approximation. Unlike the well established adaptive neural control which uses a fixed neural network structure as a complete system, in this approach the function approximation network consists of a set of bases and the number of bases can be increased when learning repeats. The nature of basis allows the continuously adaptive tuning or learning of parameters when the network undergoes a structure change, consequently offers the flexibility in tuning the network structure. The expansibility of the basis ensures the function approximation accuracy, and removes the processes in pre-setting the network size.

(4). To make a process convergent in a finite time interval, the initial condition becomes crucial because asymptotical convergence along the time horizon is no longer valid. Five different initial conditions associated with ILC are discussed. For each initial condition, the boundedness along the time horizon and asymptotical convergence along the iteration axis were exploited with rigorous analysis. Through both theoretical study and numerical examples, the Lyapunov based ILC can effectively work with sufficient robustness.

In the third part of the thesis, three different RLC approaches are proposed.

(1). For dynamic systems with unknown periodic parameters, a new RLC approach is developed. The existence of solution and learning convergence are proved with mathematical rigorousness. Robustifying the RLC approach with projection and forgetting factor has also been exploited in a systematic manner via the Lyapunov-Krasovskii functional approach.

(2). A new RLC approach is developed to handle a class of tracking control problems by making use of the repetitive nature of the control problems. The target trajectory can be any smooth periodic orbit of a nonlinear reference model. What can be learnt in RLC are either the desired periodic control signals or the lumped uncertainties which may become periodic when the system states converge to the periodic orbit of the reference model. With mathematical rigorousness we prove the existence of solution and learning convergence in a systematic manner via the Lyapunov-Krasovskii functional approach. Two robustification approaches for the nonlinear learning control with projection and forgetting factor are developed. As an extension, the integration of RLC and robust adaptive control is also exploited to address the cascaded systems without strict matching condition.

(3). As an application, an RLC approach is applied to the synchronization of two uncertain chaotic systems which contain both time varying and time invariant parametric uncertainties. The approach also deals with unknown time varying parameters having distinct periods in the master and slave systems. Using the Lyapunov-Krasovskii functional and incorporating periodic parametric learning mechanism, the global stability and asymptotic synchronization between the master and the slave systems are obtained.

List of Tables

List of Figures

Notations

Symbol	Meaning or Operation
$\forall$	for all
$\exists$	there exists
$\triangleq$	definition
$\in$	in the set
$\subset$	subset of
$\cap$	intersection of sets
$\cup$	union of sets
$\text{sign}(\star)$	signum function
$\mid \star \mid$	absolute value of a number
$\Vert \star \Vert$	Euclidean norm of vector or its induced matrix norm
$\Vert \star \Vert_2$	L^2-norm
I	an identity matrix
A^T	the transpose of A
$\lvert y(t)\rvert_s$	$\sup\limits_t \lvert y(t)\rvert$, for any scalar y
$\Vert \mathbf{y}(t)\Vert_s$	$\sup\limits_t \Vert \mathbf{y}(t)\Vert$, for any vector $\mathbf{y}$
$\Vert \cdot \Vert_T$	extended L^2-norm, defined as $\Vert \cdot \Vert_T \triangleq \frac{1}{T}\int_0^T \Vert \cdot \Vert^2 d\tau$
$\Vert \mathbf{z}_i \Vert_m$	$\max\{\lvert z_{j,i}\rvert_s : j = 1, ..., n+i\}$ for $\mathbf{z}_i = (z_{1,i}, ..., z_{n+i,i})^T$
λ_A	the minimum eigenvalue of the matrix A
$\mathcal{C}([a,b];\mathcal{R}^m)$	space of continuous functions from $[a,b]$ to $\mathcal{R}^m$
$\mathcal{C}^1([a,b];\mathcal{R}^m)$	space of continuously differentiable functions from $[a,b]$ to $\mathcal{R}^m$
$\mathcal{C}^n_{PT}([a,b];\mathcal{R}^m)$	space of $n-$order continuously differentiable and periodic functions
	with periodicity T : $\mathbf{f}(t) = \mathbf{f}(t-T)$ and the mapping $\mathbf{f} : [a,b]$ to $\mathcal{R}^m$
$\mathbf{f}_\ominus$	$\mathbf{f}(t-T)$

Chapter 1

Introduction

1.1 Background and Motivation

Learning control mainly aims at improving the system performance via directly updating the control input, either repeatedly over a fixed finite time interval, or repetitively (cyclically) over an infinite time interval. Moreover, there are two kinds of non-repeatable problems encountered in learning control: non-repeatability of a motion task and non-repeatability of a process. Many learning control methods have been proposed in the past two decades, among them three predominant are direction learning control (Xu, 1997*b*), (Xu, 1998), iterative learning control (Arimoto *et al.*, 1984*a*), (Lee and Bien, 1997), (Moore, 1998), (Chen and Wen, 1999), (Sun and Wang, 2001), (French and Phan, 2000) and (Chien and Yao, 2004), and repetitive control (Hara *et al.*, 1988), (Messner *et al.*, 1991), (Owens *et al.*, 1999), (Longman, 2000).

1.1.1 Direct Learning Control (DLC)

Generally speaking, there are two kinds of non-repeatable problems encountered in learning control: non-repeatability of a motion task and non-repeatability of a process. The non-repeatable motion task could be shown through the following example: an XY-table draws two circles with the same period but different radii. The non-repeatability of a process could be due to the nature of system such as welding different parts in a manufacturing line. Without loss of generality, we refer to these two kinds of problems as non-repeatable control problems which result in extra difficulty when a learning control scheme is to be applied.

From the practical point of view, non-repeatable learning control is very important and indispensable. In order to deal with non-repeatable learning control problems, it is needed to explore the inherent relations of different motion trajectory patterns. The resulting learning control scheme might be both plant-dependent and trajectory-dependent. On the other hand, since learning control task is essentially to drive the system tracking the given trajectories, the inherent spatial and speed relationships among distinct motion trajectories actually provide useful information. Moreover, in spite of the variations in the trajectory patterns, the underlying dynamic properties of the controlled system remain the same. Therefore, it is possible for us to deal with non-repeatable learning control problems. A control system may have plenty of prior control knowledge obtained through all the past control actions although they may correspond to different plants or different tasks. These control profiles are obviously correlated and contain a lot of important information about the system itself. In order to effectively utilize these prior control knowledge and explore the possibility of solving non-repeatable learning control problem, direction learning control schemes were proposed by (Xu, 1997*b*), (Xu, 1998).

Direct Learning Control is defined as the direct generation of the desired control profile from existing control inputs without any repeated learning. The ultimate goal of DLC is to fully utilize all the pre-stored control profiles and eliminate the time consuming iteration process thoroughly, even though these control profiles may correspond to different motion patterns and be obtained using different control methods. In this way, DLC provides a new kind of feedforward compensation, which differs from other kinds of feedforward compensation methods. A feedforward compensator hitherto is constructed in terms of the prior knowledge with regard to the plant structural or parametric uncertainties. Its effectiveness therefore depends on whether a good estimation or guess is available for these system uncertainties. In contrast with the conventional ones, DLC scheme provides an alternative way: generating a feedforward signal by directly using the information of past control actions instead of the plant parameter estimation. Another advantage of DLC is, that it can be used where repetitive operation may not be permitted.

DLC problems can be classified into the following several sub-categories:

1. Direct learning of trajectories with the same time period but different magnitude scales which can be further classified into the following two categories,

i) DLC learning of trajectories with single magnitude scale relations.

ii) DLC learning of trajectories with multiple magnitude scale relations.

2. Direct learning of trajectories with the same spatial path but different time scales. It can also be classified into two sub-categories:

i) DLC learning of trajectories with linear time scale relation.

ii) DLC learning of trajectories with nonlinear time scale mapping relations.

3. Direct learning of trajectories with variations in both time and magnitude scales.

4. Direct learning of plants with inherent relationship of two plants before and

after the switch, though both plants may be partially unknown to us.

A typical example of non-uniform task specifications can be illustrated as follows: a robotic manipulator draws circles in Cartesian space with the same radius but different periods, or on the contrary, draws circles with the same period but different radii as shown in Figure 1.1.

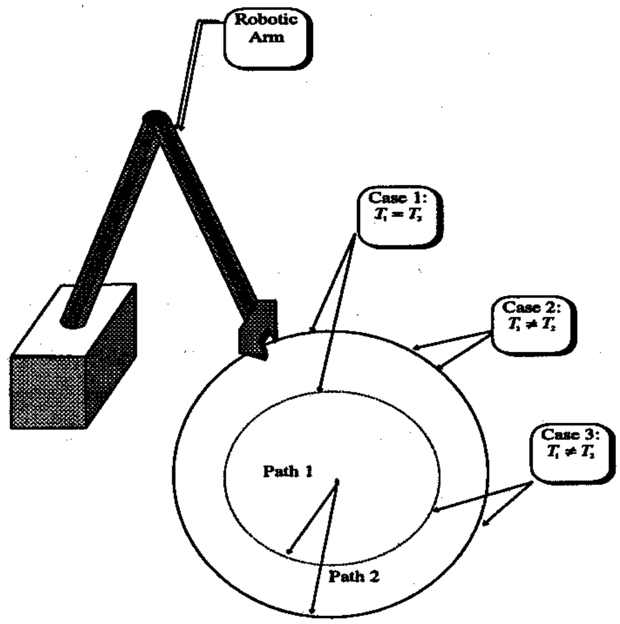

Figure 1.1. Classifications of DLC Schemes

The features of the direct learning methods are:

1. rather accurate and sufficient prior control information are required;

2. be able to learn from different motion trajectories;

3. be able to learn from different plants;

4. no need of repetitive learning because the desired control input can be calculated directly.

Therefore DLC can be regarded as an alternate for the existing learning control schemes under certain condition.

1.1.2 Iterative Learning Control (ILC)

Iterative learning control was firstly proposed by Arimoto (Arimoto *et al.*, 1984*a*). After that, many research work has been carried out in this area and a lot of theories and systematic approaches have been developed for a large variety of linear or nonlinear systems to deal with repeated tracking control problems or periodic disturbance rejection problems. Iterative learning control (ILC) has been proposed and developed as a kind of contraction mapping approach to achieve perfect tracking under the repeatable control environment which implies a repeated exosystem in a finite time interval with a strict initial resetting condition, (Arimoto *et al.*, 1984*b*), (Sugie and Ono, 1991), (Moore, 1993), (Chien, 1996), (Owens and Munde, 1996), (Xu, 1997*a*), (Park *et al.*, 1998), (Chen *et al.*, 1999), (Sun and Wang, 2002), (Xu and Tan, 2002*b*), etc. Recently new ILC approaches based on Lyapunov function technology (Qu, 2002), (Qu and Xu, 2002) and Composite Energy Function (CEF) (Xu and Tan, 2000), (Xu, 2002*b*) have been developed to complement the contraction mapping based ILC. For instance, by means of CEF based ILC, we can extend the system nonlinearities from global Lipschitz continuous to non-global Lipschitz continuous (Xu and Tan, 2000), extend target trajectories from uniform to non-uniform ones (Xu and Xu, 2002), remove the requirement on the strict initial resetting condition (Xu *et al.*, 2000), deal with time varying and norm bounded system uncertainties (Xu, 2002*b*), and incorporate nonlinear optimality (Xu and Tan, 2001), etc. ILC has been widely applied to mechanical systems such as robotics, electrical systems such as servomotors, chemical systems such as batch realtors, as well as aerodynamic systems, etc. ILC has been applied to both motion control and process control areas such as wafer process, batch reactor control, IC welding process, industrial robot control on assembly line, etc (Oh *et al.*, 1988), (Naniwa and Arimoto, 1991), (Fu and Barford, 1992), (Kuc *et al.*, 1991), (Zilouchian, 1994), (Zhang *et al.*, 1994), (Lucibello, 1996), (Lee and Lee, 1997), (Kim and Ha, 1999) and (Lee and Lee, 2000). Learning control system can enjoy the advantage of system repetition to improve the performance over the entire learning cycle.

The main strategy of ILC is to learn inputs that generate required outputs from a dynamical system by repeated trials and updating of control inputs from iteration to iteration. Though numerous methodologies of ILC have been proposed, they could be clearly classified based on the system input updating law. The main features of the existing iterative learning methods are:

1. little prior knowledge about the system is required;

2. only effective for single motion trajectory;

3. repeated learning process is needed.

Iterative learning control and direct learning control are actually functioning in a somewhat complementary manner.

The block diagram of a typical iterative learning control system is shown in Figure 1.2

In Figure 1.2, $y_r(t)$ is the desired output trajectory of the plant and $u_0(t)$ is the initial input signal for the first iteration. The target of the ILC controller is to make the output of the plant to track the desired output trajectory perfectly. The ILC system shown in Figure 1.2 consists of a previous cycle feedback (PCF) and a current cycle feedback (CCF). The controller adopts certain control algorithm, and the output of the controller is sent to the plant as input of next iteration cycle.

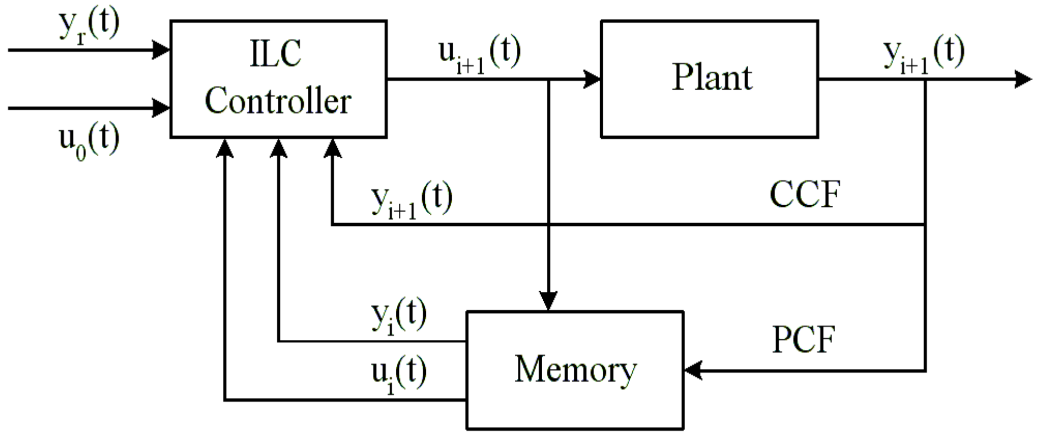

Figure 1.2. Block diagram of Iterative learning controller

Up to now, there are many approaches which can be employed to analyze ILC convergence property such as contraction mapping and energy function. Contraction mapping method is a systematic way of analyzing learning convergence. The global Lipschitz condition is a basic requirement which limits its extending to more general class of nonlinear systems. Moreover, generally the contraction mapping design only cares the tracking convergence along learning horizon, while the system stability, which is an important factor in system control, is ignored. Therefore, energy function based ILC convergence analysis is widely applied for nonlinear systems. The development of ILC focuses on several problems: the direct transmission term becomes singular; the control directions are unknown; the perfect initial resetting may not be obtainable; the dynamic system has unknown nonlinear uncertainties.

Applying the contraction mapping method, we often consider the following dynamical system

$$\begin{aligned} \dot{\mathbf{x}}(t) &= f(\mathbf{x}(t), u(t), y(t), t), \\ y(t) &= g(\mathbf{x}(t), u(t), t), \end{aligned} \tag{1.1}$$

where $t \in [0,\ T]$, $f(\cdot)$ and $g(\cdot)$ satisfy the Global Lipschitz continuity condition. This model includes a large variety of nonlinear dynamic systems with non-affine-in-input factors. Although many of existing problems have been widely studied by virtue of contraction mapping methods, it is still a challenging and open problem in ILC when the direct feed-through term becomes singular at a number of points.

Unlike the contraction mapping method, where the output tracking is considered, CEF method is concerned with the state tracking. By the latter method, more general nonlinear dynamic systems can be addressed. As a relatively new topic, CEF method brings out some open issues that need to be studied:

There are some problems in the development of CEF method.

1. A constantly challenging mission for control society is on dealing with dynamic systems in the presence of unknown nonlinearities. Consider the following simple affine dynamics

$$\dot{\mathbf{x}}(t) \quad = \quad f(t, \mathbf{x}(t)) + bu(t), \tag{1.2}$$

where u is the system input. Over the past five decades, numerous control strategies have been developed according to the scenarios associated with the structure and prior knowledge of $f(t, \mathbf{x})$. If $f(t, \mathbf{x})$ can be parameterized as the product of unknown time invariant parameters and known nonlinear functions, adaptive control and adaptive learning are most suitable. If $f(t, \mathbf{x})$ cannot be parameterized but its upperbounding function $\bar{f}(t, \mathbf{x})$ is known *a priori*, robust control or robust learning control (Tan and Xu, 2003) characterized by high gain feedback is pertinent. In the past decade, intelligent control methods using function approximation, such as neural network, fuzzy network or wavelet network, have been widely studied, which open a new avenue leading to more powerful control solutions as well as better control performance. The most profound feature of those function approximation lies in that the nonparametric function $f(\mathbf{x})$ is given a representation in a parameter space, with an artificially constructed function approximation network, e.g. RBF (radial basis function) network, MLP (multilayer perception) network, etc. Note that the artificially constructed network consists of known nonlinear functions, hence the control problem renders into an analogy as adaptive control or learning control: need only to cope with unknown time invariant parameters. This accounts for the popularity of function approximation based control, in particular neural control in recent advances (Narendra and Parthasarathy, 1990), (Hunt *et al.*, 1992), (Levin and Narendra, 1996), (Sanner and Slotine, 1992), (Polycarpou, 1996), (Seshagiri and Khalil, 2000), (Ge and Wang, 2002) and (Huang *et al.*, 2003).

2. Some works based on CEF have studied the performing tracking control with a priori knowledge of control directions, i.e., the sign of b is known. It is an extremely difficult and challenging control problem when the control directions are unknown. Up to now, there are mainly two ways to address the problem. One way is to incorporate the technique of Nussbaum-type "gains" into the control design. The first result was proposed by Nussbaum (Nussbaum, 1983), and later extended to adaptive control systems (Ryan, 1991), (Ye and Jiang, 1998) *et al*. Another way is to directly estimate unknown parameters involved in the control directions (Mudgett and Morse, 1985), (Brogliato and Lozano, 1992), (Brogliato and Lozano, 1994), (Kaloust and Qu, 1995), *et al*.

3. To make a process convergent in a finite time interval, the initial condition becomes crucial because asymptotical convergence along the time horizon is no longer valid. Iterative learning control (ILC) based on contraction mapping requires the identical initial condition (*i.i.c.*) in order to achieve a perfect tracking (Arimoto *et al.*, 1984*b*; Sugie and Ono, 1991; Ahn *et al.*, 1993; Xu and Tan, 2003). The robustness of contraction based ILC has been studied (Arimoto *et al.*, 1991; Lee and Bien, 1991; Porter and Mohamed, 1991*b*; Porter and Mohamed, 1991*a*; Heinzinger *et al.*, 1992; Saab, 1994), and several algorithms were proposed for ILC without *i.i.c.* (Park and Bien, 2000; Sun and Wang, 2002; Chen *et al.*, 1999). Recently, new ILC approaches based on CEF method (Xu and Tan, 2003; Xu and Tan,

2002*a*; Qu, 2002; Jiang and Unbehauen, 2002; Tayebi, 2004) have been developed to complement the contraction mapping based ILC in the sense that local Lipschitz nonlinearities can be taken into consideration. Majority of those approaches also require the identical initial condition. In practical applications, the perfect initial resetting may not be obtainable. That motivates us to study initial conditions for this class of ILC.

1.1.3 Repetitive Learning Control (RLC)

In practice there exists another kind of tracking control problems: the desired output trajectory or the unknown time-varying uncertainties are periodic for $t \in [0, \infty)$. Any periodic signal with period T can be generated by the time-delay systems as shown in Figure 1.3 with an appropriate initial function.

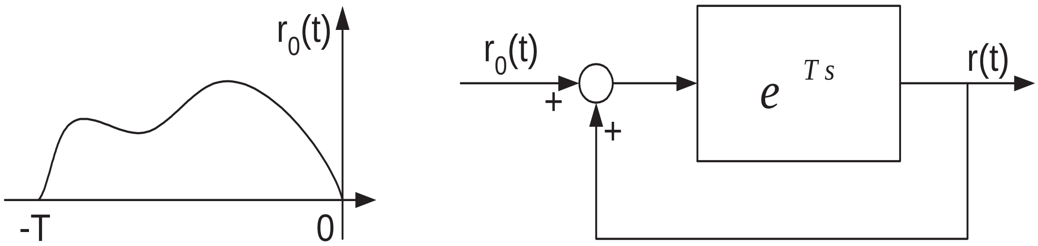

Figure 1.3. Generator of periodic signal

In contrast to ILC which has been applied to the finite time interval, the repetitive control focus on the infinite time interval. The repetitive control has been mainly applied to servo problems for LTI (linear time invariant) systems to track periodic references and reject periodic disturbances. The concept of repetitive control was first proposed in (Hara *et al.*, 1988) for LTI systems and the convergence analysis was conducted in frequency domain using small gain theorem. In (Rogers and Owens, 1992) and (Owens *et al.*, 1999), the stability analysis was conducted in the form of differential-difference equations for linear repetitive processes. In (Longman, 2000), some design issues were exploited for linear repetitive control. In (Messner and Bodson, 1995), an adaptive feedforward control using internal model equivalence was developed, which deals with LTI systems with an exogenous disturbance consisting of a finite number of sinusoidal functions, and the adaptation mechanism estimates the constant unknown coefficients.

The extension of repetitive control to nonlinear dynamics has also been exploited. In (Messner *et al.*, 1991), the learning control has been applied to identify and compensate for a nonlinear disturbance function which is represented as an integral of a predefined kernel function multiplied by an unknown influence function that is state independent. In (Vecchio *et al.*, 2003), a kind of adaptive learning control scheme was proposed for a class of feedback linearizable systems to track a periodic reference, and the problem can be converted into the learning of a finite number of Fourier coefficients. In (Dixon *et al.*, 2003), the repetitive learning control is applied to a class of nonlinear systems with matched periodic disturbance. Since the periodic disturbance is a time function, it can also be treated as an unknown periodic coefficient under the framework of adaptive control (Xu, 2004). Note that, above mentioned learning control schemes require the plant to be parameterizable and what is aimed is asymptotic convergence along the time horizon, hence they may also be regarded as some kinds of nonlinear adaptive control under the generalized framework of adaptive control theory. In (Cao and Xu, 2001), a repetitive learning control scheme was developed for nonlinear dynamics without parameterization. Nonlinear robust control is used together with the repetitive learning mechanism, hence it requires the upper bound knowledge of the lumped uncertaities.

Under the present theoretical framework of repetitive control, it would be difficult to deal with plants with unknown nonlinear components that are not parameterizable. It is necessary to seek a new learning control strategy, which is able to use the simple but effective delay-based mechanism to carry out the repetitive learning, meanwhile is able to deal with lumped nonlinear unknowns.

It has been shown that many well-known chaotic systems, including Duffing oscillator, R össler system, Chua's circuits, etc., can be transformed into the form of nonlinear dynamical systems with either unknown constant parameters or unknown time-varying factors. Adaptive control methods can well handle chaotic systems with unknown constant parameters (Wang and Ge, 2001*a*) and (Wang and Ge, 2001*b*). On the other hand, the learning control method (Song *et al.*, 2002) has been applied to chaotic systems in the presence of time-varying uncertainties with a uniform periodicity. The classical adaptive updating law and the repetitive learning law are used jointly for systems with both multi-period time-varying and time invariant parameters. Generally speaking, the classical adaptive updating law does not work for time varying parameters. The repetitive learning control law, on the other hand, does not perform as well as classical adaptive updating law for time invariant parameters due to smoothness problem.

1.2 Objectives and Contributions of the Thesis

In this thesis, the research is focused on developing new learning control approaches for linear and nonlinear dynamic systems. The main contributions lie in the following aspects: A new DLC approach is proposed for a class of linear time varying, uncertain, and switched systems; Two ILC approaches are designed by adding a forgetting factor and incorporating a time varying learning gain for a class of linear systems in the presence of input singularity, which is incurred by the singularities of the system direct transmission term; A new ILC approach is constructed with both differential and difference updating laws to deal with a class of nonlinear systems without a priori knowledge of control directions; A constructive function approximation approach is proposed for adaptive learning control which handles finite interval tracking problems; For ILC approaches, five different initial conditions are studied to disclose the inherent relationship between each initial condition and corresponding learning convergence (or boundedness) property; Two new RLC approaches are proposed for systems with either periodic unknown parameters or non-parametric uncertainties; A new learning control approach for synchronization of two uncertain chaotic systems is presented. The contributions of the thesis are summarized in Table 1.1.

Table 1.1 The contribution of the thesis

Dynamic System (Plant)		Control Methods	Convergence Analysis
Linear time-varying (LTV) switch systems		DLC	Perfect tracking
LTV system with input singularity (singular direct feed-through term)		ILC based on Contraction mapping	Uniformly bound
Nonlinear system with parametric uncertainty	Unknown control direction	ILC based on Lyapunov functional	$\|\ \|_T$ convergence
	Five different initial conditions	ILC based on Lyapunov functional	1. Point-wise convergence; 2. Subsequence convergence; 3. $\|\ \|_T$ convergence
	Known control direction	RLC based on Lyapunov functional	$\|\ \|_T$ convergence
Nonlinear system with non-parametric uncertainty		ILC based on wavelet network	Subsequence convergence
		RLC based on Lapunov-Krasovskii functional	$\|\ \|_T$ convergence
Chaotic systems		RLC based on Lapunov-Krasovskii functional	$\|\ \|_T$ convergence

In details, the contributions of this thesis are as follows:

1. In Chapter 2, a DLC approach for a class of switched systems is proposed. The objective of direct learning is to generate the desired control profile for

a newly switched system without any feedback, even if the system may have uncertainties. This is achieved by exploring the inherent relationship between any two systems before and after a switch. The new method is applicable to a class of linear time varying, uncertain, and switched systems, when the trajectory tracking control problem is concerned. Singularity problem and trajectory switch problem are also considered.

2. In Chapter 3, a challenging and open problem: how to design a suitable ILC approach in the presence of input singularity, is addressed. Considering two typical types of input singularities, ILC approaches are revised accordingly by adding a forgetting factor and incorporating a time varying learning gain, in the sequel guarantee ILC approaches to be contractible. Using Banach fixed point theorem, the output sequence can either enter and remain ultimately in a designated neighborhood of the target trajectory, or bounded by a class $\mathcal{K}$ function.

3. In Chapter 4, by incorporating a Nussbaum-type function, a new ILC approach is constructed with both differential and difference updating laws to explore the possibility of designing a suitable iterative learning control system without a priori knowledge of the control directions. The new approach can warrant a L_T^2 convergence of the tracking error sequence along the iteration axis, in the presence of time-varying parametric uncertainties and local Lipschitz nonlinearities.

4. In Chapter 5, a new constructive function approximation approach is proposed for adaptive learning control which handles finite interval tracking problems. Unlike the well established adaptive neural control which uses a fixed neural network structure as a complete system, in the method the function approximation network consists of a set of bases and the number of bases can be increased when learning repeats. The nature of basis allows

the continuously adaptive tuning or learning of parameters when the network undergoes a structure change, consequently offers the flexibility in tuning the network structure. The expansibility of the basis ensures the function approximation accuracy, and removes the processes in pre-setting the network size. Two classes of system unknown nonlinear functions, either in $\mathcal{L}^2(R)$ or a known upperbound, are taken into consideration. With the help of Lyapunov method, the existence of solution and the convergence property of the proposed adaptive learning control system, are analyzed rigorously.

5. In Chapter 6, five different initial conditions associated with ILC are discussed. For each initial condition, the boundedness along the time horizon and asymptotical convergence along the iteration axis were exploited with rigorous analysis. Through both theoretical study and numerical examples, the Lyapunov based ILC can effectively work with sufficient robustness.

6. In Chapter 7, a new RLC approach is developed for systems with unknown periodic parameters. With mathematical rigorousness the existence of solution and learning convergence are proved. Robustifying the nonlinear learning control with projection and forgetting factor is also been exploited in a systematic manner via the Lyapunov-Krasovskii functional approach.

7. In Chapter 8, a new RLC approach is developed to handle a class of tracking control problems by use of the repetitive nature of the control problems. The target trajectory can be any smooth periodic orbit of a nonlinear reference model. What can be learnt in RLC are either the desired periodic control signals or the lumped uncertainties which may become periodic when the system states converge to the periodic orbit of the reference model. With mathematical rigorousness the existence of solution and learning convergence can be proved in a systematic manner via the Lyapunov-Krasovskii functional approach. Two robustification schemes for the nonlinear learning control

with projection and forgetting factor are developed. As an extension, the integration of RLC and robust adaptive control is also exploited to address the cascaded systems without strict matching condition.

8. In Chapter 9, a learning control approach for synchronization of two uncertain chaotic systems is presented. Global stability and asymptotic synchronization are achieved for chaotic systems with both time-varying and time invariant parametric uncertainties.

Chapter 2

Direct Learning Control Design for a Class of Linear Time-varying Switched Systems

2.1 Introduction

System switches may arise in many practical processes. Many hybrid systems consist of multiple subsystems and switch according to certain switching laws. In a complex system, many types of changes may be encountered, e.g., faults in the system, changes in subsystem dynamics and changes in system parameters.

In general, complex systems operate in multiple environments which may change abruptly from one context to another (Ezzine and Haddad, 1989), (Liberzon and Morse, 1999), (Ye *et al.*, 1998), (Ji and Chizeck, 1988), (Loparo *et al.*, 1987). One typical switch type engineering system is an electrical circuit with many relay components, which has been widely applied in the field of power electronics (Sira-Ramirez, 1991). Any on-off switch of a relay may give rise to the change in the system topology and parameters. Other examples of switch systems can be found in power systems (Williams and Hoft, 1991), building air-condition, communication network, etc.

Drawing more attentions recently, switched systems have been widely investigated, mainly focusing on the system properties such as controllability, observability, and stability (Sun and Zheng, 2001), (Stanford and L. T. Conner, 1980) and (Branicky, 1998). In this chapter, we concentrate on the tracking control problem for switched systems. Traditionally control system design has been based on a single fixed model of the system. When the system switches, there is a need to re-design the closed-loop so as to generate the desired control input profiles. In addition, it takes time for the system to converge, or eliminate the tracking error asymptotically. Can we find a quick and easy way to generate the desired control signals without re-designing the controller, and the target trajectory can be followed from the beginning?

Direct Learning Control (DLC) method was proposed by (Xu, 1997*b*), (Xu, 1998) to directly generate the desired control profile from pre-stored control inputs. DLC works for a fixed system with switched target trajectories, that is, the desired control profile can be directly generated, even if the new trajectory may be different from any existing trajectories tracked previously. The key idea of DLC is to use the inherent relationships between the new and existing trajectories, hence a feed-forward control can be implemented. In this chapter, we will extend the same idea to system switches.

When a system switches, often we know the topological variation before and after the switch. For instance, we are able to known the change of a network when an on-off operation of a relay occurs, though we may not know the details of the network components. In other words, we may have some inherent relationship of two systems before and after the switch, though both systems may be partially unknown to us. If we can acquire a sufficient number of such relationships associated with switches, there is a possibility of directly generating the new control profile with respect to the new system. It is worthwhile to point out, that a new control system may have plenty of prior control knowledge obtained through all the past control actions although they may correspond to the different systems. In this chapter, we will focus on a class of time-varying switched systems, show how we can fully utilize the pre-stored control information, and explore the conditions assuring a direct learning of the desired control input profile.

The chapter is organized as follows. Section 2.2 states the control problem for a class of linear time-varying switched systems. Section 2.3 provides a new direct learning scheme to obtain the desired control profile. Section 2.4 presents an illustrative example.

2.2 Problem statement

Consider the switched systems given by the following equations:

$$\dot{\mathbf{x}}_i(t) \quad = \quad A_i(t)\mathbf{x}_i(t) + B_i(t)\mathbf{u}_i(t), \tag{2.1}$$

where $\mathbf{x}_i = [x_{1,i} \ \cdots \ x_{n,i}]^T$ is the $i-$th system state vector. $A_i(t)$, $B_i(t) \in R^{n\times n}$, are unknown time-varying matrices. $B_i(t)$ is full rank for $\forall t \in [0,\ T]$, $i \in \mathcal{N}$, where $[0,\ T]$ is the tracking period.

The control objective is to find the control input for a tracking control trajectory $\mathbf{x}_d$ over the given time period $t \in [0,\ T]$, where $\mathbf{x}_d(t) = [x_{1,d}(t) \ \cdots \ x_{n,d}(t)]^T$ represents the desired system state trajectory. For the switched systems, a new control system may have plenty of prior control knowledge obtained through all the past control actions although they may correspond to different systems. In this chapter, in order to effectively utilize all the prior control knowledge so as to remove the iterative learning process, we will propose a new DLC scheme for the class of linear time-varying switched systems.

Assumption 2.1. *Any two consecutive switched systems have the following relations:*

$$A_i(t) = K_{i-1}A_{i-1}(t), \quad B_i(t) = M_{i-1}B_{i-1}(t), \tag{2.2}$$

where K_j, M_j, $j = 1, 2, \cdots$, are all constant matrices, and M_j is of full rank.

Assumption 2.2. *There are at least $N = 2n^2$ known tracking control input profiles $\mathbf{u}_i(t)$ available.*

Now consider the new system

$$\dot{\mathbf{x}}_{N+1}(t) \;=\; A_{N+1}(t)\mathbf{x}_{N+1}(t) + B_{N+1}(t)\mathbf{u}_{N+1}(t). \tag{2.3}$$

Our control objective is also to find the control input $\mathbf{u}_d(t)$ to track the trajectory $\mathbf{x}_d(t)$. When $\mathbf{x}_{N+1}(t) = \mathbf{x}_d(t)$, $\mathbf{u}_d(t)$ should be

$$\mathbf{u}_d(t) = B^{-1}_{N+1}(t)\dot{\mathbf{x}}_d(t) - B^{-1}_{N+1}(t)A_{N+1}(t)\mathbf{x}_d(t). \tag{2.4}$$

Note that because $A_{N+1}(t)$ and $B_{N+1}(t)$ are unknown, the control input $\mathbf{u}_d(t)$ cannot be calculated directly from the above equation.

According to the relations (2.2), we have

$$\begin{aligned} A_i(t) &= K_{i-1}A_{i-1}(t) = \cdots = \prod_{j=1}^{i-1} K_{i-j}A_1(t), \\ B_i(t) &= M_{i-1}B_{i-1}(t) = \cdots = \prod_{j=1}^{i-1} M_{i-j}B_1(t). \end{aligned} \tag{2.5}$$

Let

$$\begin{aligned} C(t) &\triangleq B_1^{-1}(t), \quad D_i \triangleq \left(\prod_{j=1}^{i-1} M_{i-j}\right)^{-1}, \\ E_i &\triangleq \left(\prod_{j=1}^{i-1} M_{i-j}\right)^{-1} \cdot \left(\prod_{j=1}^{i-1} K_{i-j}\right), \end{aligned} \tag{2.6}$$

where

$$\begin{aligned} C(t) &= \begin{bmatrix} \mathbf{c}_1(t) \\ \vdots \\ \mathbf{c}_n(t) \end{bmatrix}, \\ D_i &= [\mathbf{d}_{1,i} \cdots \mathbf{d}_{n,i}], \quad E_i = [\mathbf{e}_{1,i} \cdots \mathbf{e}_{n,i}]. \end{aligned} \tag{2.7}$$

To facilitate the derivation of DLC in subsequent section, the following lemma is given.

Lemma 2.1. *For any matrix* $\Phi \in R^{n\times n} = \begin{bmatrix} \boldsymbol{\phi}_1 & \cdots & \boldsymbol{\phi}_n \end{bmatrix}^T \in R^{n\times n}$ *and* $\Gamma = [\boldsymbol{\gamma}_1 \cdots \boldsymbol{\gamma}_n] \in R^{n\times n}$*, the following relation holds:*

$$\Phi\Gamma = \sum_{j=1}^{n}\sum_{k=1}^{n} \Gamma^{jk}\Phi^{jk} \tag{2.8}$$

where

$$\begin{aligned} \Gamma^{jk} &= \begin{bmatrix} \overbrace{\mathbf{0} \;\cdots\; \mathbf{0} \quad \underbrace{\boldsymbol{\gamma}_k}_{j^{th}\ column} \quad \mathbf{0} \;\cdots\; \mathbf{0}}^{n\quad columns} \end{bmatrix}^T, \\ \Phi^{jk} &= \begin{bmatrix} \overbrace{\mathbf{0} \;\cdots\; \mathbf{0} \quad \underbrace{\boldsymbol{\phi}_j^T}_{k^{th}\ column} \quad \mathbf{0} \;\cdots \mathbf{0}}^{n\quad ccolumns} \end{bmatrix}. \end{aligned}$$

Proof. See the Appendix A.1. □

2.3 Derivation of the DLC Scheme

In this section, the DLC scheme for the switched systems will be given. For convenience, let

$$\begin{aligned}
C^{jk}(t) &\triangleq \begin{bmatrix} \overbrace{\mathbf{0} \;\cdots\; \mathbf{0} \quad \underbrace{\mathbf{c}_j^T}_{k^{th}\ column} \quad \mathbf{0} \;\cdots \mathbf{0}}^{n\quad columns} \end{bmatrix}, \\
D_i^{jk} &\triangleq \begin{bmatrix} \overbrace{\mathbf{0} \;\cdots\; \mathbf{0} \quad \underbrace{\mathbf{d}_{k,i}}_{j^{th}\ column} \quad \mathbf{0} \;\cdots\; \mathbf{0}}^{n\quad columns} \end{bmatrix}^T, \\
E_i^{jk} &\triangleq \begin{bmatrix} \overbrace{\mathbf{0} \;\cdots\; \mathbf{0} \quad \underbrace{\mathbf{e}_{k,i}}_{j^{th}\ column} \quad \mathbf{0} \;\cdots\; \mathbf{0}}^{n\quad columns} \end{bmatrix}^T,
\end{aligned} \tag{2.9}$$

where $\mathbf{c}_j$, $\mathbf{d}_{k,i}$ and $\mathbf{e}_{k,i}$ are given by (2.7),

$$\begin{aligned}
\bar{D}_i &\triangleq \begin{bmatrix} D_i^{11} & \cdots & D_i^{1n} & \cdots & D_i^{n1} & \cdots & D_i^{nn} \end{bmatrix} \\
&= \begin{bmatrix} \mathbf{d}_{1,i}^T & \cdots & \mathbf{d}_{n,i}^T & \cdots & \mathbf{0} & \cdots & \mathbf{0} \\ \mathbf{0} & \cdots & \mathbf{0} & \cdots & \mathbf{0} & \cdots & \mathbf{0} \\ \vdots & \ddots & \vdots & \vdots & \vdots & \ddots & \vdots \\ \mathbf{0} & \cdots & \mathbf{0} & \cdots & \mathbf{d}_{1,i}^T & \cdots & \mathbf{d}_{n,i}^T \end{bmatrix}, \\
\bar{E}_i &\triangleq \begin{bmatrix} E_i^{11} & \cdots & E_i^{1n} & \cdots & E_i^{n1} & \cdots & E_i^{nn} \end{bmatrix} \\
&= \begin{bmatrix} \mathbf{e}_{1,i}^T & \cdots & \mathbf{e}_{n,i}^T & \cdots & \mathbf{0} & \cdots & \mathbf{0} \\ \mathbf{0} & \cdots & \mathbf{0} & \cdots & \mathbf{0} & \cdots & \mathbf{0} \\ \vdots & \ddots & \vdots & \vdots & \vdots & \ddots & \vdots \\ \mathbf{0} & \cdots & \mathbf{0} & \cdots & \mathbf{e}_{1,i}^T & \cdots & \mathbf{e}_{n,i}^T \end{bmatrix},
\end{aligned} \tag{2.10}$$

$$R = \begin{bmatrix} \bar{D}_1 & \bar{E}_1 \\ \vdots & \vdots \\ \bar{D}_i & \bar{E}_i \\ \vdots & \vdots \\ \bar{D}_N & \bar{E}_N \end{bmatrix}, \tag{2.11}$$

and

$$S = \begin{bmatrix} \bar{D}_{N+1} & \bar{E}_{N+1} \end{bmatrix}. \tag{2.12}$$

The following assumption is necessary.

Assumption 2.3. *The N learned switched systems are correlated with the new system (2.3) in such a way that*

$$R_1 = \begin{bmatrix} \mathbf{d}_{1,1}^T & \cdots & \mathbf{d}_{n,1}^T & \mathbf{e}_{1,1}^T & \cdots & \mathbf{e}_{n,1}^T \\ \vdots & \ddots & \vdots & \vdots & \ddots & \vdots \\ \mathbf{d}_{1,N}^T & \cdots & \mathbf{d}_{n,N}^T & \mathbf{e}_{1,N}^T & \cdots & \mathbf{e}_{n,N}^T \end{bmatrix} \tag{2.13}$$

is full rank.

Lemma 2.2. *The rank of the matrix R is equivalent to the rank of the matrix R_1, where R and R_1 are defined in (2.11) and (2.13) respectively.*

Proof. See the Appendix A.2. □

The main result is given in the following theorem.

Theorem 2.1. *The desired control input $\mathbf{u}_d(t)$ with respect to the system (2.3) can be directly obtained using N past control inputs according to the following relation:*

$$\mathbf{u}_d(t) = SR^{-1} \begin{bmatrix} \mathbf{u}_1(t) \\ \vdots \\ \mathbf{u}_N(t) \end{bmatrix}, \tag{2.14}$$

where $\mathbf{u}_i(t)$, $i = 1, \cdots, N$, is the known desired control input profile of the $i-th$ switched system (2.1), S and R are defined in (2.12) and (2.11) respectively.

Proof. Because $B_i(t)$ is of full rank, then (2.1) can be written as follows:

$$\mathbf{u}_i(t) = B_i^{-1}(t)\dot{\mathbf{x}}_d - B_i^{-1}(t)A_i(t)\mathbf{x}_d. \tag{2.15}$$

According to the Assumption 2.1, substituting the equation (2.5) and (2.6) into (2.15), the following relation can be obtained:

$$\begin{aligned}\mathbf{u}_i(t) &= B_i^{-1}(t)\dot{\mathbf{x}}_d - B_i^{-1}(t)A_i(t)\mathbf{x}_d \\ &= B_1^{-1}(t)\left(\prod_{j=1}^{i-1} M_{i-j}\right)^{-1}\dot{\mathbf{x}}_d - B_1^{-1}(t)\left(\prod_{j=1}^{i-1} M_{i-j}\right)^{-1} \times \left(\prod_{j=1}^{i-1} K_{i-j}\right) A_1(t)\mathbf{x}_d \\ &= C(t)D_i\dot{\mathbf{x}}_d - C(t)E_iA_1(t)\mathbf{x}_d, \end{aligned} \tag{2.16}$$

According to Lemma 2.1, the following relation exists:

$$\begin{aligned}\mathbf{u}_i(t) &= \left(\sum_{j=1}^{n}\sum_{k=1}^{n} D_i^{jk}C^{jk}(t)\right)\dot{\mathbf{x}}_d - \left(\sum_{j=1}^{n}\sum_{k=1}^{n} E_i^{jk}C^{jk}(t)\right)A_1(t)\mathbf{x}_d \\ &= \sum_{j=1}^{n}\sum_{k=1}^{n} D_i^{jk}C^{jk}(t)\dot{\mathbf{x}}_d - \sum_{j=1}^{n}\sum_{k=1}^{n} E_i^{jk}C^{jk}(t)A_1(t)\mathbf{x}_d \end{aligned} \tag{2.17}$$

where C^{jk}, D_i^{jk} and E_i^{jk} are defined in (2.9). By rearranging the above equation, we have

$$\begin{aligned}\mathbf{u}_i(t) &= \sum_{j=1}^{n}\sum_{k=1}^{n}\left(D_i^{jk}C^{jk}(t)\dot{\mathbf{x}}_d - E_i^{jk}C^{jk}(t)A_1(t)\mathbf{x}_d\right) \\ &= \begin{bmatrix} \bar{D}_i & \bar{E}_i \end{bmatrix}\mathbf{z}(t) \end{aligned} \tag{2.18}$$

where $\bar{D}_i$ and $\bar{E}_i$ are given in (2.10) and

$$\begin{aligned} &\mathbf{z}(t) \triangleq \begin{bmatrix} \mathbf{z}_1(t) & \mathbf{z}_2(t) \end{bmatrix}^T \\ &\mathbf{z}_j(t) \triangleq \begin{bmatrix} z_j^{11}(t) & \cdots & z_j^{2n}(t) & \cdots & z_j^{n1}(t) & \cdots & z_j^{nn}(t) \end{bmatrix}^T, \qquad j = 1, 2, \\ &z_1^{ml}(t) \triangleq C^{ml}(t)\dot{\mathbf{x}}_d, \\ &z_2^{ml}(t) \triangleq C^{ml}(t)A_1(t)\mathbf{x}_d, \quad m, l = 1, \cdots, n. \end{aligned} \tag{2.19}$$

The vector $\mathbf{z}(t)$, which is a set of unknown basis functions and switch-irrelevant, can be learned directly in a point-wise manner with the known coefficient matrix D_i^{jk}, E_i^{jk} and control input $\mathbf{u}_i(t)$.

From Assumption 2.2, we know that there are $N = 2n^2$ previously stored control profiles. (2.18) can be rewritten in a form: $\mathbf{u}(t) = R\mathbf{z}(t)$, where

$$\mathbf{u}(t) = \begin{bmatrix} \mathbf{u}_1^T(t) & \cdots & \mathbf{u}_i^T(t) & \cdots & \mathbf{u}_N^T(t) \end{bmatrix}^T,$$

R and $\mathbf{z}(t)$ represent the known scaling matrix and unknown basis respectively.

From Lemma 2.2 and Assumption 2.3, R is of full rank. Therefore, $\mathbf{z}(t)$ can be obtained as

$$\mathbf{z}(t) = R^{-1}\mathbf{u}(t). \tag{2.20}$$

Similar to (2.17), utilizing the denotation (2.6) and the definition of $\mathbf{z}(t)$ in (2.19) , (2.4) can be rewritten as

$$\begin{aligned}
\mathbf{u}_d(t) &= B_1^{-1}(t)\left(\prod_{j=1}^{N} M_{N+1-j}\right)^{-1}\dot{\mathbf{x}}_d - B_1^{-1}(t)\left(\prod_{j=1}^{N} M_{N+1-j}\right)^{-1} \\
&\quad \times \left(\prod_{j=1}^{N} K_{N+1-j}\right) A_1(t)\mathbf{x}_d \\
&= C(t)D_{N+1}\dot{\mathbf{x}}_d - C(t)E_{N+1}A_1(t)\mathbf{x}_d \\
&= \sum_{j=1}^{n}\sum_{j=1}^{n} D_{N+1}^{jk}C^{jk}(t)\dot{\mathbf{x}}_d - \sum_{j=1}^{n}\sum_{j=1}^{n} E_{N+1}^{jk}C^{jk}(t)A_1(t)\mathbf{x}_d \\
&= S\mathbf{z}(t)
\end{aligned} \tag{2.21}$$

where S is given in (2.12) .

Substituting (2.20), the new desired control input is directly achieved. This completes the proof. □

Remark 2.1. *We can extend the above result to more generic circumstances:*

1. *If the matrix R_1 is singular, extra control input profiles should be added to improve the rank condition of R_1. The DLC scheme remains almost the same, and the terms $\mathbf{z}(t)$ can be obtained in the sense of Least Squares.*

2. *On the other hand, if the matrices K_i, M_i, $i \in N$, are all diagonal, it is sufficient to use $2n$ known tracking control input profiles to generate the desired control input profile.*

3. *If the target trajectories also switch at different operation cycles, the DLC scheme is still applicable with some minor modifications.*

Remark 2.2. *Although we assume constant K_i , M_i, $i \in \mathcal{N}$, the proposed DLC can be extended straightforward to time-varying cases, as long as the ranking condition is satisfied.*

2.4 Illustrative Example

In this section, the proposed DLC scheme is applied to the linear switched systems for illustrative purpose. The switched systems are described by the following equation:

$$\dot{\mathbf{x}}_i(t) \quad = \quad A_i(t)\mathbf{x}_i(t) + B_i(t)\mathbf{u}_i(t) \tag{2.22}$$

where

$$A_1(t) = \begin{bmatrix} -1 & 1 \\ -1 & 2 \end{bmatrix}, \quad B_1(t) = \begin{bmatrix} 1 & \sin(t) \\ -1 & 2 \end{bmatrix},$$

and $\mathbf{x}_i(t)$, $\mathbf{u}_i(t)$ are the $i-$th system states to be controlled and control inputs respectively.

Let the desired trajectory is $\mathbf{x}_d(t) = \begin{bmatrix} \sin(t) \\ \cos(t) \end{bmatrix}, \quad t \in [0,\ 2\pi]$, the systems switch eight times and K_i and M_i, $i = 1, 2, \cdots, 7$, are given as follows:

$$K_1 = \begin{bmatrix} 1 & 0.2 \\ 1 & -1 \end{bmatrix}, \qquad K_2 = \begin{bmatrix} 0 & -1 \\ 6.67 & 0.33 \end{bmatrix},$$
$$K_3 = \begin{bmatrix} 1.88 & 0.13 \\ -1.06 & 0.06 \end{bmatrix}, \quad K_4 = \begin{bmatrix} 4.25 & 3.5 \\ -2.5 & -7 \end{bmatrix},$$
$$K_5 = \begin{bmatrix} 0.19 & -0.23 \\ 0.33 & 0.17 \end{bmatrix}, \quad K_6 = \begin{bmatrix} 4.56 & -0.89 \\ 0.67 & 2.33 \end{bmatrix},$$
$$K_7 = \begin{bmatrix} 0.41 & 0.08 \\ -0.16 & 0.42 \end{bmatrix}, \quad M_1 = \begin{bmatrix} -1 & 2 \\ 0.9 & 1 \end{bmatrix},$$
$$M_2 = \begin{bmatrix} 0.29 & 1.43 \\ -1.75 & 2.5 \end{bmatrix}, \quad M_3 = \begin{bmatrix} 0.78 & -0.44 \\ 1.56 & 1.11 \end{bmatrix},$$
$$M_4 = \begin{bmatrix} -0.14 & 0.14 \\ -0.86 & 0.36 \end{bmatrix}, \quad M_5 = \begin{bmatrix} 13 & -4 \\ 0 & 1 \end{bmatrix},$$
$$M_6 = \begin{bmatrix} 1 & 0 \\ -0.08 & 0.69 \end{bmatrix}, \quad M_7 = \begin{bmatrix} 0.4 & 0.3 \\ 0.33 & 2.33 \end{bmatrix}.$$

Now consider the following new system

$$\dot{\mathbf{x}}_9(t) = A_9(t)\mathbf{x}_9(t) + B_9(t)\mathbf{u}_9(t), \tag{2.23}$$

where K_8 and M_8 are

$$K_8 = \begin{bmatrix} 0.43 & 0.12 \\ -3 & 1.91 \end{bmatrix}, \quad M_8 = \begin{bmatrix} 1.2 & 0.66 \\ -0.27 & 0.65 \end{bmatrix}. \tag{2.24}$$

Since the control input profiles of the previous eight systems are known *a priori*, that is, $\mathbf{u}_1, \mathbf{u}_2, \mathbf{u}_3$,$\mathbf{u}_4$, $\mathbf{u}_5$,$\mathbf{u}_6$,$\mathbf{u}_7$ and $\mathbf{u}_8$ are available, by applying the proposed DLC scheme in Theorem 1, the control input $\mathbf{u}_d(t)$ is obtained directly. Simulation results are presented in Figure 2.1. From the figure it can be observed that the directly learned control input profiles are exactly the same as the desired ones. The DLC scheme can successfully learn and generate the desired control signals from the switched system.

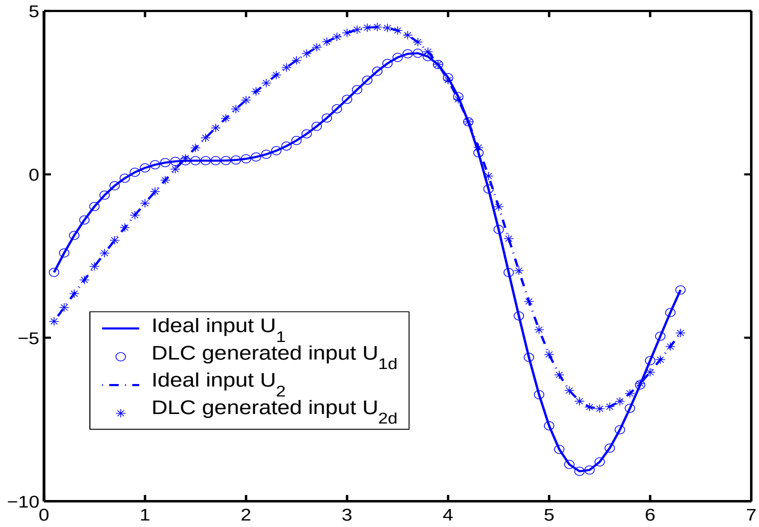

Figure 2.1. DLC obtained control input

2.5 Conclusion

To solve the trajectory tracking problem for a class of switched systems, a new direct learning control method is proposed and verified. The new DLC control allows the full use of pre-obtained control signals, in the sequel generates the desired control profile for a newly switched system. We have shown that the new method is applicable to a class of linear time-varying systems with uncertainties. Simulation results further confirm the effectiveness of the new method.

Chapter 3

Fixed Point Theorem based Iterative Learning Control for Linear Time-varying Systems with Input Singularity

3.1 Introduction

Iterative learning control (ILC) has been intensively studies in near the past two decades (Arimoto *et al.*, 1984*a*), (Kuc *et al.*, 1992), (Jang *et al.*, 1995), (Moore, 1998), (Chien, 1998), (Longman, 1998), (Wang, 1998), (Chen *et al.*, 1998), (Ghash and Paden, 2002), (Xu and Tan, 2002*c*). From a rigorous mathematical viewpoint, ILC is a kind of function approximation based on contraction mapping and fixed point theorem. The well known ILC updating law, usually linking two consecutive iterations, provides a specific approximation operator that ensures the convergence. Meanwhile, under the Global Lipschitz continuity condition, the uniqueness of the control input, which achieves the perfect tracking, is guaranteed. However, contraction mapping based ILC requires a nonsingular direct feed-through term between the system input and output.

Iterative Learning Control (ILC), based on contraction mapping, is a kind of output tracking control and the relative degree of the system needs to be zero, i.e., the direct feedthrough term must be nonsingular in general for all ILC problems. On the other hand, the identical initial condition plus global Lipschitz condition (GLC) will ensure the boundedness of the state in any finite time interval. Therefore ILC will work and achieve perfect output tracking in a finite interval, regardless of the stability and controllability of the state dynamics. For instance, even if there exists an unstable and uncontrollable mode, by virtue of the identical initial condition, together with the GLC, the mode will not incur any finite escape time phenomenon. On the other hand, owing to the algebraic relation between the input and output, output variables can be directly manipulated by inputs, regardless of any finite state values. This is also a major advantage of ILC.

In this chapter we consider a very challenging and open problem in ILC: the direct feed-through term becomes singular at a number of points. Since the learnability condition is violated at those points, we need to look for alternative contraction mapping approaches according to various types of singularities, such that the fixed point theorem is still applicable. Two types of singularities are considered in this chapter. In the first situation, the direct feed-through term does not changes signs (the control direction) on the two sides of a singular point. It is relatively easy to address this type of singularities. We need only to do a very minor modification to a conventional ILC operator by adding a forgetting factor close to unity. The focus of this part of work is to exhibit two important issues: 1) the revised contraction mapping generates a control input sequence converging to a unique fixed point uniformly, and 2) this fixed point warrants the system output to ultimately and uniformly enter a designated neighborhood of the target trajectory. In this simple ILC design, we do not need to know the locations of the singular points.

It is however much more difficult to handle the second type of singularities: on two sides of a singular point the direct feed-through term changes the sign. Thus it is necessary to know when a second type singularity occurs, and how does the sign changes. In addition to the forgetting factor, which alone is insufficient in such circumstance, we further incorporate the sign changes into the revised ILC operator. We can demonstrate that 1) the revised ILC operator is contractible and the control input sequence converges uniformly to a unique fixed point, 2) the system enters a designated neighborhood of the target trajectory except for a number of sub-intervals centered about the second type singular points, and 3) within each sub-interval the tracking error is bounded by a class $\mathcal{K}$ function of a quantity which specifies the bound of the designated neighborhood.

Due to the extreme difficulty in dealing with input singularities, in this chapter we focus on linear time varying (LTV) systems. Nevertheless the results can be extended straightforward to a class of nonlinear dynamic systems.

This chapter is organized as follows. Section 3.2 gives problem formulation and some preliminaries. Sections 3.3 and 3.4 address the two types of singularities respectively. Section 3.5 presents an illustrative example.

3.2 Problem Formulation and Preliminaries

Consider a class of LTV systems described by

$$\begin{aligned} \dot{\mathbf{x}}(t) &= A(t)\mathbf{x}(t) + \mathbf{b}(t)u(t) \qquad \mathbf{x}(a) = \mathbf{x}_a \\ y(t) &= \mathbf{c}(t)\mathbf{x}(t) + d(t)u(t), \end{aligned} \tag{3.1}$$

where $t \in [a,b] \triangleq I$, a and b are finite positive constants, $A(t) \in \mathcal{C}^0(I, \mathcal{R}^{n\times n})$, $\mathbf{b}(t) \in \mathcal{C}^0(I, \mathcal{R}^{n\times 1})$, $\mathbf{c}(t) \in \mathcal{C}^1(I, \mathcal{R}^{1\times n})$ and $d(t) \in \mathcal{C}^1(I, \mathcal{R})$ respectively. Here we adopt the notations $\mathcal{R} = (-\infty, \infty)$, and $\mathcal{C}^p(I, \mathcal{R}^n)$ the space of continuous functions ($p = 0$) and the space of continuously differentiable functions ($p = 1$), which map the interval I into $\mathcal{R}^n$.

Since ILC works under a repeatable control environment, the identical initial condition is assumed

Assumption 3.1.

$$\mathbf{x}_i(a) = \mathbf{x}_a \quad for \quad i = 1, 2, \cdots. \tag{3.2}$$

where the subscript i denotes the i^{th} iteration.

From the continuity of $A(t)$, $\mathbf{b}(t)$, the smoothness of $\mathbf{c}(t)$ and $d(t)$, and the finite interval, there exist finite positive constants β_A, $\beta_{\mathbf{b}}$, $\beta_{\mathbf{c}}$, and β_d such that $\|A(t)\|_s = \beta_A$, $\|\mathbf{b}(t)\|_s = \beta_{\mathbf{b}}$, $\|\mathbf{c}(t)\|_s = \beta_{\mathbf{c}}$, and $|d(t)|_s = \beta_d$ for $\forall t \in I$. Here $\|\cdot\|$ represents the infinity norm for a vector, and the induced norm for a matrix. $\|\cdot\|_s$ represents the supreme norm for a vector valued or matrix valued function defined in I, i.e. $\|\cdot\|_s = \sup_{t\in I} \|\cdot\|$. When a scalar is concerned, the infinity norm or the function norm renders to $|\cdot|$ or $|\cdot|_s$. To facilitate the subsequent discussions, a time weighted norm is also defined

$$\|\cdot\|_\lambda = \sup_{t\in I} e^{-\lambda(t-a)}\|\cdot\|$$

where λ must be a finite constant so that the function norm can be well defined over the interval I.

Give a target trajectory $y_r(t) \in \mathcal{C}^1(I, \mathcal{R})$. The objective of ILC is to construct an appropriate contraction operator, that generates a convergent input sequence $u_i(t)$ leading to a unique fixed point $u_r(t)$ for $\forall t \in I$. In the sequel the output sequence $y_i(t)$, driven by $u_i(t)$, converges to $y_r(t)$. Such a contraction mapping has been proposed in (Arimoto *et al.*, 1984*a*), and is valid when the system direct feedthrough term is nonsingular, i.e. $|d(t)|_s \geq \alpha > 0$. The objective of this chapter, is to extend the ILC to a more general case where $|d(t)| = 0$ for a number of points $t \in I$.

The following properties will be used in subsequent sections.

Property 3.1. *$\mathcal{C}^p(I, \mathcal{R}^n)$ and $\mathcal{C}^p(I, \mathcal{R}^n, \|\cdot\|_\lambda)$, $p = 0, 1$, are both Banach spaces. In fact it is well known that $\mathcal{C}(I, \mathcal{R}^n)$ is a Banach space. From the norm equivalence*

$$e^{-\lambda(b-a)}\|\cdot\|_s \leq \|\cdot\|_\lambda \leq \|\cdot\|_s$$

it is immediately obvious that $\mathcal{C}(I, \mathcal{R}^n, \|\cdot\|_\lambda)$ is also a Banach space.

Property 3.2. *Let $\mathcal{T}$ be a contraction operator in a Banach space $\mathcal{X}$. Then according to Banach fixed point theorem*
1) $\mathcal{T}$ has a unique fixed point $x^* \in \mathcal{X}$, and
2) for any initial approximation $x_a \in \mathcal{X}$, the sequence of successive approximations

$$x_{i+1} = \mathcal{T}(x_i), \qquad k = 0, 1, 2, \cdots \tag{3.3}$$

converges to x^*.

Property 3.3. *For any finite positive constants q and γ, there exists a finite value of λ such that the following relationship holds for the dynamic system (3.1)*

$$|q\mathbf{c}(\mathbf{x}_1 - \mathbf{x}_2)|_\lambda \leq \frac{\gamma}{2}|u_1 - u_2|_\lambda. \tag{3.4}$$

This property is derived as follows. From Assumption 3.1 the identical initial condition, substituting the dynamics (3.1) and applying Gronwall Lemma, we have

$$\begin{aligned}
\|\mathbf{x}_1(t) - \mathbf{x}_2(t)\| &= \|\int_a^t [A(\tau)\mathbf{x}_1(\tau) + B(\tau)u_1(\tau) - A(\tau)\mathbf{x}_2(\tau) - B(\tau)u_2(\tau)]d\tau\| \\
&\leq \beta_A \int_a^t \|\mathbf{x}_1(\tau) - \mathbf{x}_2(\tau)\|d\tau + \beta_{\mathbf{b}} \int_a^t |u_1(\tau) - u_2(\tau)|d\tau \\
&\leq \beta_A \int_a^t \|\mathbf{x}_1(\tau) - \mathbf{x}_2(\tau)\|d\tau + \beta_{\mathbf{b}} \int_a^t e^{\lambda(\tau-a)}|u_1 - u_2|_\lambda d\tau \\
&\leq \frac{e^{\lambda t} - 1}{\lambda}\beta_{\mathbf{b}} e^{\beta_A t}|u_1 - u_2|_\lambda \\
\Rightarrow \|\mathbf{x}_1 - \mathbf{x}_2\|_\lambda &\leq \frac{1 - e^{-\lambda(b-a)}}{\lambda}\beta_{\mathbf{b}} e^{\beta_A(b-a)}|u_1 - u_2|_\lambda
\end{aligned} \tag{3.5}$$

Therefore

$$|q\mathbf{c}(\mathbf{x}_1 - \mathbf{x}_2)|_\lambda \leq \frac{1 - e^{-\lambda(b-a)}}{\lambda} q\beta_{\mathbf{c}}\beta_{\mathbf{b}} e^{\beta_A(b-a)}|u_1 - u_2|_\lambda.$$

Let

$$\frac{1 - e^{-\lambda(b-a)}}{\lambda} q\beta_{\mathbf{c}}\beta_{\mathbf{b}} e^{\beta_A(b-a)} \leq \frac{\gamma}{2},$$

by ignoring $e^{-\lambda(b-a)}$ we have

$$\lambda \geq \frac{2q\beta_{\mathbf{c}}\beta_{\mathbf{b}} e^{\beta_A(b-a)}}{\gamma}.$$

This property has been widely used for ILC convergence analysis in the presence of the system dynamics. Generally speaking, the impact from the system state dynamics to the system output, i.e. the $\mathbf{c}(t)\mathbf{x}(t)$ term in the output equation, can be handled in two ways. If the tracking interval is sufficiently short such that the direct feed-through term is dominant in terms of the supreme norm, we can derive the contraction property directly using the supreme norm (Lee and Bien, 1997). However, when the tracking interval is larger, the dynamic impact may grow exponentially to reach the scale of $e^{\beta_A(b-a)}$, and become dominant in the output equation if the supreme norm is still applied. In such case, the time weighted norm will have to be used to suppress the exponential growth. Since a monotonically convergent sequence in $\|\cdot\|_\lambda$ may actually grow up for a finite number of iterations in terms of the supreme norm, a frequently raised question is whether $\|\cdot\|_s$ can be applied even if the tracking interval is large. Unfortunately this is an extremely difficult problem as it requires the capability of controlling the transient behavior in iteration domain. As we know, in much of control literature the transient behavior of a control system is still open in general. Transient improvement can be expected only when more of the system knowledge is available, such as the use of Markov parameters to describe the system dynamics (French and Phan, 2000), or learning in state space (Xu, 2002a). While in the presence of input singularities, convergence analysis becomes extremely difficult, let alone the transient behavior. Thus throughout this chapter, the convergence analysis is made in the sense of the time weighted norm.

3.3 ILC for the First Type of Singularities

The existence of the input singularity prevents an ILC operator from generating an ultimately uniformly convergent sequence. The system learnability condition is violated at any t where $d(t) = 0$. The best we can expect is for the tracking error to uniformly enter a prespecified neighborhood below

$$|y_r(t) - y_i(t)|_s \leq \epsilon \tag{3.6}$$

as $i \to \infty$. ϵ specifies the error metric bound.

Surprisingly, as we will show in this section, the following simple ILC operator can do the job well

$$u_{i+1}(t) = (1 - \gamma)u_i(t) + q[y_r(t) - y_i(t)] \tag{3.7}$$

where γ is a constant satisfying $0 < \gamma \ll 1$, and q is a learning gain. γ plays the role of a forgetting factor. Note that this learning law is equivalent to the following operator

$$\mathcal{T}[u(t)] = (1 - \gamma)u(t) + q[y_r(t) - y(t)]. \tag{3.8}$$

In fact this ILC operator frequently appears in the ILC literature, and the sole purpose of the forgetting factor is to robustify learning in the presence of exogenous perturbations. Our main contribution in this section is to demonstrate that the same ILC operator is also valid for input singularities. Namely, (3.7) remains a contraction operator in the presence of singularities, and achieves the desired tracking bound (3.6) by deliberately choosing the parameters γ and q. For simplicity we will omit the argument t in subsequent derivations wherever no confusion arises.

Theorem 3.1. *The operator (3.8) warrants a convergent sequence u_i to a unique fixed point $u^* \in \mathcal{C}^1(I, \mathcal{R}, \|\cdot\|_\lambda)$, and achieves the desired performance (3.6) for any $\epsilon > 0$, when the control parameters are chosen to be $0 < q \leq \dfrac{2\beta_{u^*}}{2\epsilon + \beta_d\beta_{u^*}}$ and $0 < \gamma \leq \dfrac{\epsilon q}{\beta_{u^*}} \leq \dfrac{2\epsilon}{2\epsilon + \beta_d\beta_{u^*}}$. Here $\beta_{u^*} \geq |u^*|_s$ is a constant.*

Proof. When $u \in \mathcal{C}^1(I, \mathcal{R}, \|\cdot\|_\lambda)$, according to the system dynamics (3.1), $\mathbf{x} \in \mathcal{C}^1(I, \mathcal{R}^{n\times 1}, \|\cdot\|_\lambda)$. In the sequel $y \in \mathcal{C}^1(I, \mathcal{R}, \|\cdot\|_\lambda)$. From (3.8), we can conclude that $\mathcal{T}$ is an operator which maps the elements of the Banach space $\mathcal{C}^1(I, \mathcal{R}, \|\cdot\|_\lambda)$ into itself.

Now we prove that $\mathcal{T}$ given in (3.8) is a contraction operator in the space $\mathcal{C}^1(I, \mathcal{R}, \|\cdot\|_\lambda)$. $\forall\ u_1,\ u_2 \in \mathcal{C}^1(I, \mathcal{R}, \|\cdot\|_\lambda)$, we have

$$\begin{aligned} |\mathcal{T}(u_1) - \mathcal{T}(u_2)|_\lambda &= |(1-\gamma-qd)(u_1-u_2) - q\mathbf{c}(\mathbf{x}_1-\mathbf{x}_2)|_\lambda \\ &\leq |1-\gamma-qd|_s|u_1-u_2|_\lambda + |q\mathbf{c}(\mathbf{x}_1-\mathbf{x}_2)|_\lambda. \end{aligned} \tag{3.9}$$

From Property 3.3 there exists a finite λ such that (3.4) holds. On the other hand, we can derive, with the selected control parameters,

$$|1-\gamma-qd|_s \leq 1-\gamma. \tag{3.10}$$

In fact, it is obvious that $1-\gamma-qd \leq 1-\gamma$ because of $q > 0$ and $d(t) \geq 0$. On the other hand,

$$\begin{aligned} 1-\gamma-qd &\geq 1-\frac{2\epsilon}{2\epsilon+\beta_{u^*}\beta_d}-\frac{2\beta_{u^*}\beta_d}{2\epsilon+\beta_{u^*}\beta_d} \\ &= -\frac{\beta_{u^*}\beta_d}{2\epsilon+\beta_{u^*}\beta_d}=-(1-\frac{2\epsilon}{2\epsilon+\beta_{u^*}\beta_d}) \\ &\geq -1+\gamma. \end{aligned}$$

Therefore, we have

$$\begin{aligned} |\mathcal{T}(u_1)-\mathcal{T}(u_2)|_\lambda &\leq |1-\gamma-qd|_s|u_1-u_2|_\lambda+|q\mathbf{c}(\mathbf{x}_1-\mathbf{x}_2)|_\lambda \\ &\leq (1-\frac{\gamma}{2})|u_1-u_2|_\lambda, \end{aligned}$$

that is, $\mathcal{T}$ is indeed a contraction operator in the Banach space $\mathcal{C}^1(I,\mathcal{R},\|\cdot\|_\lambda)$. According to Banach fixed point theorem, we can immediately conclude that $\mathcal{T}$ has a unique fixed point $u^*(t)\in\mathcal{C}^1(I,\mathcal{R},\|\cdot\|_\lambda)$, and for any initial approximation $u_0(t)\in\mathcal{C}^1(I,\mathcal{R},\|\cdot\|_\lambda)$, the sequence of successive approximations $u_{i+1}=\mathcal{T}(u_i)$ converges to u^*.

The remaining question is, will u^* enable the corresponding system output y to enter the neighborhood (3.6)? Since $u^*=\mathcal{T}(u^*)$, substituting $u=u^*$ into (3.8), taking the supreme norm on both sides, further substituting q and the upperbound of γ, we finally have

$$|y_r(t)-y(t)|_s=\frac{\gamma|u^*(t)|_s}{q}\leq\epsilon. \tag{3.11}$$

This completes the proof. □

Remark 3.1. *The smaller the parameter ϵ, the closer is $y(t)$ to the objective trajectory $y_r(t)$. This means that we can specify the tracking accuracy by choosing an appropriate value for the design parameter ϵ.*

Remark 3.2. *In determining the control parameters γ and q, we need the bounding knowledge of u^* which may not be known to us. In practice we can partially address this problem in two ways, either using a sufficiently large estimate of u^*,* *or updating the bound* $\beta_{u^*} = \max\{|u_i|_s, |u_{i-1}|_s\}$*. If* u^* *is over-estimated, the prespecified tracking accuracy will certainly be achieved. If* u^* *is under-estimated, the tracking error will still be uniformly bounded, but may not be in the prespecified neighborhood (6).*

3.4 ILC for the Second Type of Singularities

When the direct feed-through term $d(t)$ changes signs across the singular points, more knowledge is needed about $d(t)$. In the first place it is necessary to know the sign changes of $d(t)$, so that the control direction determined by $q(t)d(t)$ can remain the same. One way is to let $q(t) = sign[d(t)]$. However a discontinuous learning control will give rise to tremendous problems in both theoretical analysis and real time implementation. Thus we consider a smooth control gain $q(t) \in \mathcal{C}^1(I, \mathcal{R})$, which ensures $q(t)d(t) \geq 0$. Here the control parameter q is no longer a constant, but a time varying gain. The ILC law is

$$u_{i+1}(t) = (1-\gamma)u_i(t) + q(t)[y_r(t) - y_i(t)] \tag{3.12}$$

or expressed equivalently by an ILC operator

$$\mathcal{T}[u(t)] = (1-\gamma)u(t) + q(t)[y_r(t) - y(t)]. \tag{3.13}$$

In the following theorem, we prove that (3.13) defines a contractible operator.

Theorem 3.2. *The operator (3.13) warrants a convergent sequence* u_i *to a unique fixed point* $u^* \in C(I, \mathcal{R}, \|\cdot\|_\lambda)$*, when the control parameters are chosen as* $0 \leq |q(t)| \leq q_m \leq \dfrac{2\beta_{u^*}}{2\epsilon + \beta_d\beta_{u^*}}$ *and* $0 < \gamma \leq \dfrac{q_m\epsilon}{\beta_{u^*}} \leq \dfrac{2\epsilon}{2\epsilon + \beta_d\beta_{u^*}}$.

Proof. Comparing (3.13) with (3.8), or comparing (3.12) with (3.7), the only difference is the replacement of a constant q by a time varying $q(t)$. Therefore analogous to the proof of Theorem 3.1, $\forall\ u_1,\ u_2 \in \mathcal{C}^1(I, \mathcal{R}, \|\cdot\|_\lambda)$, we have

$$|\mathcal{T}(u_1) - \mathcal{T}(u_2)|_\lambda \ \leq\ |1-\gamma-qd|_s|u_1-u_2|_\lambda + |q\mathbf{c}(\mathbf{x}_1-\mathbf{x}_2)|_\lambda. \tag{3.14}$$

If $|1-\gamma-qd|_s \leq 1-\gamma$, the ILC operator (3.13) is a contractible operator and u^* is unique.

It is obvious that $1-\gamma-q(t)d(t) \leq 1-\gamma$ because $q(t)d(t) \geq 0$. Moreover,

$$\begin{aligned} 1-\gamma-qd &\geq 1-\frac{2\epsilon}{2\epsilon+\beta_{u^*}\beta_d} - \frac{2\beta_{u^*}\beta_d}{2\epsilon+\beta_{u^*}\beta_d} \\ &= -\frac{\beta_{u^*}\beta_d}{2\epsilon+\beta_{u^*}\beta_d} = -(1-\frac{2\epsilon}{2\epsilon+\beta_{u^*}\beta_d}) \\ &\geq -1+\gamma. \end{aligned}$$

Following the discussion in Theorem 3.1, it concludes

$$|\mathcal{T}(u_1) - \mathcal{T}(u_2)|_\lambda \leq (1-\frac{\gamma}{2})|u_1-u_2|_\lambda.$$

□

Now let us discuss the tracking performance. Since $u^* = \mathcal{T}(u^*)$, from (3.13) we can derive

$$|q(t)||y_r(t)-y(t)| = \gamma|u^*(t)|. \tag{3.15}$$

It is not possible to derive the uniform boundedness property as in Theorem 3.1, because $q(t)$ goes to zero at singular points. In order to exploit the boundedness property, divide the interval I into two sets $\Omega_1 = \{t \in I : |q(t)| \geq q_m\}$ and $\Omega_2 = I - \Omega_1$. For all $t \in \Omega_1$, (3.15) can be rewritten as

$$|y_r(t)-y(t)| \leq \frac{\gamma|u^*(t)|}{q_m}. \tag{3.16}$$

Thus analogous to Theorem 3.1, $|y_r(t)-y(t)| \leq \epsilon$ for $\forall t \in \Omega_1$.

What kind of bounding property can we draw in a small interval nearby a second type singular point where $|q(t)| < q_m$? Since $q(t)$ is a design parameter, we can judiciously choose it such that Ω_2 consists of a number of open sets (neighborhoods), each covers a second type singular point with the interval length $\delta(\epsilon)$, where $\delta(\epsilon)$ is a class $\mathcal{K}$ function of ϵ, i.e. continuous, strictly increasing and $\delta(0) = 0$. For instance, we can choose $q(t) = q_m \sin \frac{\pi}{\epsilon}(t - t_s)$ nearby a singular point t_s which produces sign changes at two sides: $d(t_s^+) > 0$ and $d(t_s^-) < 0$. Then the corresponding neighborhood is an open interval $(t_s - \frac{\epsilon}{2}, t_s + \frac{\epsilon}{2})$ with the interval length $\delta(\epsilon) = \epsilon$. In the following we prove the boundedness property for any interval in Ω_2.

Corollary 3.1. *The output tracking error metric in the neighborhood of a second type singular point is a class $\mathcal{K}$ function of ϵ.*

Proof. Denote y^* and $\mathbf{x}^*$ respectively the system output and states corresponding to u^*. Define an interval $I_s = (t_s - \frac{\delta(\epsilon)}{2}, t_s + \frac{\delta(\epsilon)}{2})$. Our objective is to show $\forall t \in I_s$, the quantity $|y_r(t) - y^*(t)|$ is a class $\mathcal{K}$ function of ϵ. First consider an upper bound of the tracking error metric

$$|y_r(t) - y^*(t)| \le |y_r(t) - y_r(t_1)| + |y_r(t_1) - y^*(t_1)| + |y^*(t_1) - y^*(t)| \qquad (3.17)$$

where $t_1 = t_s - \dfrac{\delta(\epsilon)}{2}$. Note that $(t_1, t] \subset I_s$, therefore $|t - t_1| \le \delta(\epsilon)$. Since $y_r(t) \in \mathcal{C}^1(I, \mathcal{R})$, its derivative is finite in I_s. Applying the mean value theorem

$$|y_r(t) - y_r(t_1)| \le |\dot{y}_r(\xi)||t - t_1| \le \delta_1(\epsilon) \quad \xi \in (t_s, t) \subset I_s$$

where $\delta_1(\epsilon)$ is class $\mathcal{K}$ function of ϵ. We also have $|y_r(t_1) - y^*(t_1)| \le \epsilon$ because $t_1 \in \Omega_1$.

Now let us evaluate $|y^*(t_1) - y^*(t)|$ or $|y^*(t) - y^*(t_1)|$. Again applying the mean value theorem

$$|y^*(t) - y^*(t_1)| \le |\dot{y}^*(\xi)||t - t_1| \le |\dot{y}^*(\xi)|\delta(\epsilon) \quad \xi \in (t_1, t) \subset I_s.$$

Let us verify that $|\dot{y}^*(t)|$ is finite for any $t \in I_s$. In fact from $u^* \in \mathcal{C}^1(I, \mathcal{R}, \|\cdot\|_\lambda)$ and the LTV dynamics (3.1), we can conclude $\mathbf{x}^* \in \mathcal{C}^1(I, \mathcal{R}^{n\times n})$ and $\dot{u}^* \in \mathcal{C}^1(I, \mathcal{R}, \|\cdot\|_\lambda)$, hence both are bounded in the interval I_s. In addition, from $\mathbf{c} \in \mathcal{C}^1(I, \mathcal{R}^{1\times n})$ and $d \in \mathcal{C}^1(I, \mathcal{R})$, we can conclude that $\dot{\mathbf{c}}$ and $\dot{d}$ are bounded in the interval I_s. Consequently

$$\dot{y}^* = \dot{\mathbf{c}}\mathbf{x}^* + \mathbf{c}\dot{\mathbf{x}}^* + \dot{d}u^* + d\dot{u}^*$$

is bounded for $\forall t \in I_s$, and there exists a class $\mathcal{K}$ function $\delta_2(\epsilon)$ such that $|y^*(t) - y^*(t_s)| \leq \delta_2(\epsilon)$.

Finally we reach the conclusion that $|y_r(t) - y^*(t)| \leq \delta_1(\epsilon) + \epsilon + \delta_2(\epsilon)$. □

Remark 3.3. *The significance of the corollary is, we can indirectly control the tracking error nearby the singular points, by means of choosing a sufficiently small* ϵ*, although we do not know the exact bound on* Ω_2*.*

Remark 3.4. *Note that in deriving the conclusion of Theorem 3.1, we do not use any information of the derivatives of* $\mathbf{c}(t)$ *and* $d(t)$*. Thus we only need* $\mathbf{c}(t)$ *and* $d(t)$ *in* $\mathcal{C}^0$*, instead of* $\mathcal{C}^1$*. As far as the second type singularity is concerned, we only need* $\mathbf{c}(t)$ *and* $d(t)$ *belonging to* $\mathcal{C}^1$ *in the neighborhoods of* Ω_2*. It is adequate for* $\mathbf{c}(t)$ *and* $d(t)$ *to be* $\mathcal{C}^0$ *in* Ω_1*.*

Remark 3.5. *Though only a LTV system is considered, the results can be extended straightforward to a class of nonlinear systems*

$$\begin{aligned} \dot{\mathbf{x}} &= \mathbf{f}(\mathbf{x}, u, t) \quad \mathbf{x}(a) = \mathbf{x}_a \\ y &= g(\mathbf{x}, t) + d(t)u, \end{aligned}$$

with $\mathbf{f}$ *and* g *global Lipschitz continuous.*

Remark 3.6. *The above results can also be applied to D-type ILC where* $d(t) \equiv 0$ *and* $\mathbf{c}(t)\mathbf{b}(t)$ *has singularities.*

3.5 Illustrative Example

Consider the following system

$$\begin{aligned} \dot{x}(t) &= \sin(t)x(t) + u(t) \quad x(0) = 0.5 \\ y(t) &= x(t) + (1-t)u(t) \end{aligned} \tag{3.18}$$

where $\beta_d = 1$. The target trajectory is

$$y_r(t) = (t-1)^2, \quad t \in [0, 1.5]. \tag{3.19}$$

There exists a singular point of the second type at $t = 1$. Choose $\epsilon = 0.01$ and a sufficiently large $\beta_{u^*} = 10$, then $q_m \le \dfrac{2 \times 10}{2 \times 0.01 + 1 \times 10} \approx 2$, and $0 < \gamma \le \dfrac{2 \times 0.01}{1} = 0.02$. In this example we choose $\gamma = 0.0001$, $q_m = 1.5$, and $I_s = (0.995,\ 1.005)$, then a simple form of the time varying gain is

$$q(t) = \begin{cases} 1.5 & \text{if } \ t \in [0,\ 0.995] \\ 1.5 \sin \dfrac{\pi}{2} \dfrac{1-t}{0.005} & \text{if } \ t \in I_s \\ -1.5 & \text{if } \ t \in [1.005,\ 1.5]. \end{cases} \tag{3.20}$$

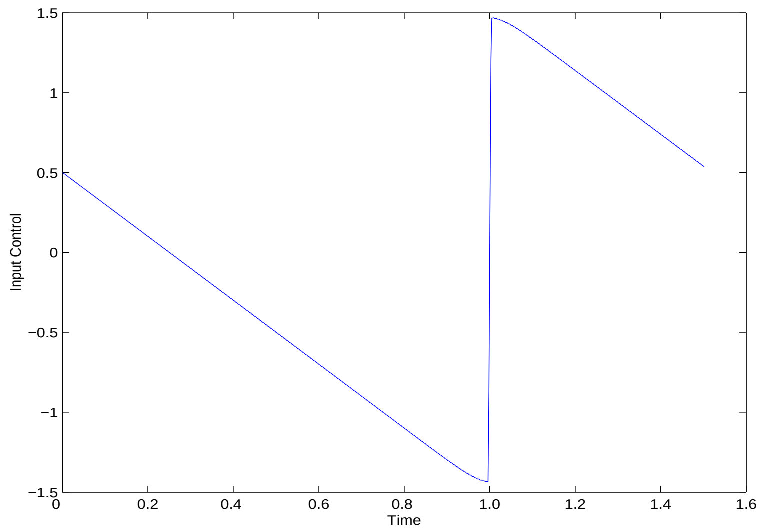

Figure 3.1. Output tracking ($i = 20$)

Figure 3.1 shows that the output $y_{20}(t)$ almost overlaps the target trajectory $y_r(t)$. Figure 3.2 show shows the difference clearly nearby the singular time point $t = 1$

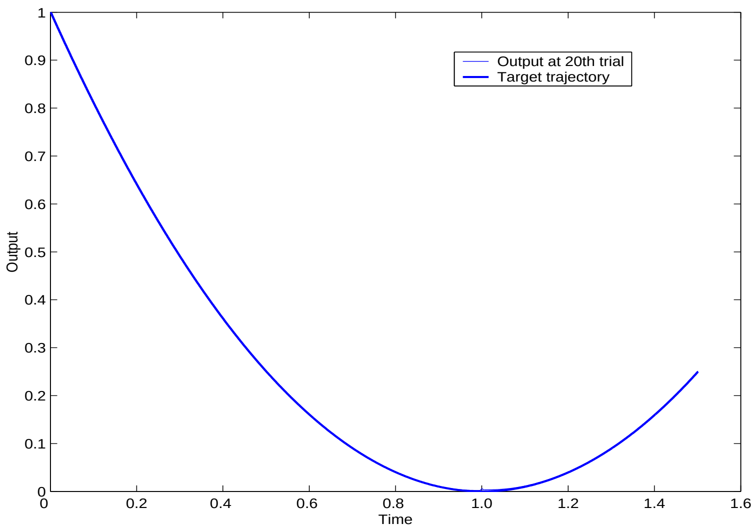

Figure 3.2. Output tracking nearby the singularity ($i = 20$)

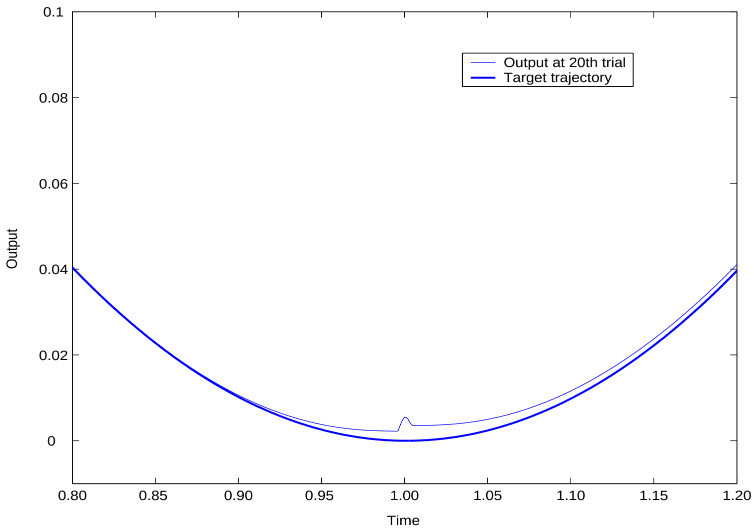

Figure 3.3. Control input ($i = 20$)

second. The tracking error is actually well below the specified bound ϵ. The control input profile is shown in Figure 3.3. The validity of the proposed ILC is confirmed.

3.6 Conclusion

In order to deal with input singularities, we present two kinds of ILC operators by adding a forgetting factor and adopting a time varying learning gain. Using Banach fixed point theorem, the proposed ILC operators ensure a convergent control input sequence approaching to a unique fixed point. In the presence of the first type of singularities, the fixed point guarantees that the system output enters and remains uniformly in a designated neighborhood of the target trajectory. While in the presence of the second type of singularities, the tracking error is bounded by a class $\mathcal{K}$ function of the designated neighborhood. The effectiveness of the ILC operators is demonstrated through an numerical example.

Chapter 4

Iterative Learning Control Design Without a Priori Knowledge of the Control Direction

4.1 Introduction

Iterative learning control (ILC) has been proposed and developed as a kind of contraction mapping approach to achieve perfect tracking under the repeatable control environment which implies a repeated trajectory over a finite time interval with the identical initialization condition (*i.i.c.*) (Arimoto *et al.*, 1984*b*; Sugie and Ono, 1991; Moore, 1993; Chien, 1996; Owens and Munde, 1996; Park *et al.*, 1998; Chen *et al.*, 1999; Sun and Wang, 2002), etc. Recently new ILC approaches based on Lyapunov function technology (Qu, 2002; Qu and Xu, 2002) and Composite Energy Function (CEF) (Xu and Tan, 2002*a*; Xu, 2002*b*) have been developed to complement the contraction mapping based ILC.

In this chapter we will show one new feature of ILC, designed based on CEF, that it can perform tracking control without *a priori* knowledge of the control direction. It is a difficult and challenging control problem when the control direction is unknown. Up to now, there are mainly two ways to address the problem. One way is to incorporate the technique of Nussbaum-type "gains" into the control design. The first result was proposed by Nussbaum (Nussbaum, 1983), and later extended to adaptive control systems (Ryan, 1991; Ye and Jiang, 1998), learning control system (Chen and Jiang, 2002). Another way is to directly estimate unknown parameters involved in the control direction (Mudgett and Morse, 1985; Brogliato and Lozano, 1992; Brogliato and Lozano, 1994; Kaloust and Qu, 1995), *et al*.

In this chapter we will adopt the first approach to deal with the unknown control direction which is determined by an unknown constant. Based on CEF, we consider the typical ILC problem: perfect tracking in finite interval. By introducing both differential and difference updating laws in the ILC mechanism, we are able to deal with systems without knowing the control direction, and in the presence of time varying parametric uncertainties associated with local Lipschitz nonlinearities. Comparing with (Chen and Jiang, 2002), the learning control scheme proposed in this chapter can be applied to more general dynamical processes with local Lipschitz nonlinearities, and system nonlinear and uncertain factors need not be uniformly bounded in the large.

The chapter is organized as follows. Section 4.2 presents the new learning control scheme. Section 4.3 exhibits the rigorous analysis of learning convergence in L^2 using CEF. Section 4.4 presents an illustrative example.

4.2 Learning Controller Design

In this section, we will consider the learning control in the repeated control environment, where the tracking task ends in a finite interval and repeats.

Consider the following uncertain nonlinear system

$$\dot{x} = \theta(t)\xi(x) + bu(t) \qquad x(0) = x_0, \tag{4.1}$$

where $\xi(x)$ is a known nonlinear function which can be local Lipschitzian, $\theta(t)$ is an unknown continuous time-varying function and $b \neq 0$ is an unknown constant parameter. The sign of b, which determines the control direction, is assumed unknown.

Consider the target trajectory generated by a reference model

$$\dot{x}_r = f(x_r, r, t), \tag{4.2}$$

where $f(x_r, r, t)$ is a known smooth function, r is a reference input which yields a bounded state $x_r(t)$ over the interval $[0,\ T]$. Define the tracking error $e(t) = x_r(t) - x(t)$, the ultimate control objective is to find a sequence of appropriate control input $u_i(t)$ $t \in [0,\ T]$ such that the system state x_i tracks the target trajectory x_r, i.e., as the learning repeats, the control system converges in L_T^2, as follows

$$\lim_{i\to\infty} \|e_i\|_T \triangleq \lim_{i\to\infty} \int_0^T e_i^2(t)dt = 0.$$

When the parameter b is known, this tracking problem has been solved in (Xu and Tan, 2002a). When b is unknown, we need to look for a new ILC approach. For this purpose the Nussbaum-type function will be used in the control law design.

Definition 4.1. $v(\cdot)$ *is an even smooth Nussbaum-type function, if the function* *has the following properties*

$$\begin{aligned}\lim_{s\to\infty}\sup\frac{1}{s}\int_0^s v(k)dk &= \infty,\\ \lim_{s\to\infty}\inf\frac{1}{s}\int_0^s v(k)dk &= -\infty.\end{aligned}\tag{4.3}$$

An example of such a continuous function is $v(k) = k^2\cos(k)$. It is clear that $v(k)$ is positive on intervals $(2n\pi, 2n\pi+\frac{\pi}{2})$ and negative on intervals $(2n\pi+\frac{\pi}{2}, 2n\pi+\frac{3\pi}{2})$, n is an integer. It is sufficient to prove that

$$\begin{aligned}\lim_{n\to\infty}\frac{1}{2n\pi+\frac{\pi}{2}}\int_0^{2n\pi+\frac{\pi}{2}} v(k)dk &= \infty,\\ \lim_{n\to\infty}\frac{1}{2n\pi+\frac{3\pi}{2}}\int_0^{2n\pi+\frac{3\pi}{2}} v(k)dk &= -\infty.\end{aligned}\tag{4.4}$$

To prove the former, we have

$$\begin{aligned}&\lim_{n\to\infty}\frac{1}{2n\pi+\frac{\pi}{2}}\int_0^{2n\pi+\frac{\pi}{2}} v(k)dk\\ &= \lim_{n\to\infty}\frac{1}{2n\pi+\frac{\pi}{2}}\int_0^{2n\pi+\frac{\pi}{2}} k^2 d\sin k\\ &= \lim_{n\to\infty}\frac{1}{2n\pi+\frac{\pi}{2}}(k^2\sin k|_0^{2n\pi+\frac{\pi}{2}} - 2\int_0^{2n\pi+\frac{\pi}{2}} k\sin kdk)\\ &= \lim_{n\to\infty}(2n\pi+\frac{\pi}{2}) - \lim_{n\to\infty}\frac{1}{n\pi+\frac{\pi}{4}}\\ &= +\infty.\end{aligned}\tag{4.5}$$

The proof of $\lim_{n\to\infty}\frac{1}{2n\pi+\frac{3\pi}{2}}\int_0^{2n\pi+\frac{3\pi}{2}} v(k)dk = -\infty$ is similar.

Associated with the Nussbaum-type function, the following property holds (Ye and Jiang, 1998).

Property 4.1. *Let $V(\cdot)$ and $k(\cdot)$ be smooth functions defined on $[t_0,\ t_f)$ with $V(t) \geq 0$, $\forall t \in [t_0,\ t_f)$, $v(\cdot)$ an even smooth Nussbaum-type function, and b a nonzero constant. If the following inequality holds:*

$$V(t) \leq \int_{t_0}^t [bv(k(\tau))+1]\dot{k}(\tau)d\tau + c, \qquad \forall t\in[t_0,\ t_f)\tag{4.6}$$

where c is an arbitrary constant, then $V(t)$*,* $k(t)$ *and* $\int_{t_0}^{t}[bv(k(\tau))+1]\dot{k}(\tau)d\tau$ *must be bounded on* $[t_0,\ t_f)$*.*

To achieve the perfect tracking result, a practical initial condition is given for each iteration as below.

Assumption 4.1. $\theta(0)=\theta(T)$, $x_i(0)=x_{i-1}(T)$. *In addition, the target trajectory* $x_r(t)$ *satisfies* $x_r(0)=x_r(T)$*.*

In most engineering systems the physical state will not jump because of the finite driving power. Hence the end of the preceding operation cycle naturally becomes the initial state of the subsequent operation cycle.

Define the learning error at the $i-$th iteration $e_i(t)=x_r(t)-x_i(t)$. Under Assumption 4.1, the error dynamics at the i-th iteration can be expressed as

$$\begin{aligned} \dot{e}_i(t) &= f(x_r,r,t)-\theta(t)\xi(x_i)-bu_i(t), \qquad \forall t\in[0,\ T] \qquad (4.7)\\ e_0(0) &= x_r(0)-x_0(0),\\ e_i(0) &= e_{i-1}(T), \qquad i\geq 1. \end{aligned}$$

The learning control mechanism is given as below:

$$\begin{aligned} u_i(t) &= v(k_i(t))z_i(t), \qquad (4.8)\\ \dot{k}_i(t) &= z_i(t)e_i(t), \qquad k_i(0)=k_{i-1}(T),\ k_0(0)=0,\\ z_i(t) &= e_i(t)+f(x_r,r,t)-\hat{\theta}_i(t)\xi(x_i), \end{aligned}$$

and the parametric updating law is $\forall t\in[0,\ T]$

$$\hat{\theta}_i(t)=\begin{cases} 0, & i=-1,\\ -\gamma_0(t)\xi(x_i)e_i(t), & i=0,\\ \hat{\theta}_{i-1}(t)-\xi(x_i)e_i(t), & i\geq 1, \end{cases} \qquad (4.9)$$

where $\gamma_0(t)$ is a continuous and strictly increasing function satisfied $\gamma_0(0) = 0$, and $\gamma_0(T) = 1$. $v(\cdot)$ is an even smooth Nussbaum-type function. For notational convenience, in subsequent context we will omit the argument t for all variables where no confusion arises, and denote $\xi(x_i)$ by ξ_i.

Now we show an alignment property associated with the quantities $\hat{\theta}_i(t)$ and $\dot{k}_i(t)$.

Property 4.2. *The learning scheme (4.8) and (4.9) ensures $\hat{\theta}_i(0) = \hat{\theta}_{i-1}(T)$ and $\dot{k}_i(0) = \dot{k}_{i-1}(T)$.*

Proof. Let us prove the first relationship by induction. For $i = 0$, from (4.9) we have $\hat{\theta}_0(0) = \hat{\theta}_{-1}(T) = 0$. Now assume that

$$\hat{\theta}_j(0) = \hat{\theta}_{j-1}(T), \text{for } j = 1, \cdots, i-1. \tag{4.10}$$

From (4.8), Assumption 4.1 and (4.10), we have

$$\hat{\theta}_i(0) = \hat{\theta}_{i-1}(0) - \xi(x_i(0))e_i(0), \tag{4.11}$$

and

$$\begin{aligned} \hat{\theta}_{i-1}(T) &= \hat{\theta}_{i-2}(T) - \xi(x_{i-1}(T))e_{i-1}(T) \\ &= \hat{\theta}_{i-1}(0) - \xi(x_i(0))e_i(0) \\ &= \hat{\theta}_i(0), \end{aligned} \tag{4.12}$$

that is, $\hat{\theta}_i(0) = \hat{\theta}_{i-1}(T)$. From (4.8), it is easy to see that $\dot{k}_i(0) = \dot{k}_{i-1}(T)$ because of $e_i(0) = e_{i-1}(T)$, $x_i(0) = x_{i-1}(T)$ and $\hat{\theta}_i(0) = \hat{\theta}_{i-1}(T)$. □

Substituting the learning control law into the error dynamics (4.7) yields

$$\begin{aligned} \dot{e}_i &= \dot{x}_r - \theta\xi_i - bu_i \\ &= \dot{x}_r - \theta\xi_i - z_i + z_i - bu_i \\ &= -e_i - (\theta - \hat{\theta}_i)\xi_i + (-bv(k_i) + 1)z_i. \end{aligned} \tag{4.13}$$

When the control direction is known *a priori*, for instance $b > 0$, the corresponding learning control law is (Xu and Tan, 2002*a*)

$$\begin{aligned} u_i &= z_i, \\ \hat{\theta}_i &= \hat{\theta}_{i-1} - \xi_i e_i. \end{aligned} \tag{4.14}$$

Without *a prioir* knowledge in the control direction, the learning control mechanism is now a mixture of differential and difference updating laws.

4.3 Learning Convergence Analysis

Now we exhibit the learning convergence property, which is summarized in the following theorem.

Theorem 4.1. *For system (4.1) under the learning control scheme (4.8) and (4.9), the learning error sequence e_i converges to zero in L_T^2.*

Proof. Define the following Lyapunov functional

$$E_i(t) = \frac{1}{2}e_i^2(t) + \frac{1}{2}\int_0^t \phi_i^2(\tau)d\tau + \frac{1}{2}\int_t^T \phi_{i-1}^2(\tau)d\tau, \tag{4.15}$$

where $\phi_i(t) = \theta(t) - \hat{\theta}_i(t)$.

The proof consists of three parts which address respectively the difference of the CEF, and the L_T^2 convergence, and the boundedness of the first iteration.

Part I: Difference of $E_i(t)$

The difference of $E_i(t)$ is

$$\begin{aligned} \Delta E_i &= E_i - E_{i-1} \\ &= \frac{1}{2}e_i^2 - \frac{1}{2}e_{i-1}^2 + \frac{1}{2}\int_0^t (\phi_i^2 - \phi_{i-1}^2)d\tau + \frac{1}{2}\int_t^T (\phi_{i-1}^2 - \phi_{i-2}^2)d\tau. \end{aligned} \tag{4.16}$$

Substituting the control law (4.8) and the error dynamics (4.13), the first term on the right hand side is

$$\begin{aligned}\frac{1}{2}e_i^2 &= \int_0^t e_i\dot{e}_i d\tau + \frac{1}{2}e_i^2(0)\\ &= \int_0^t e_i[-e_i - (\theta-\hat{\theta}_i)\xi_i + (-bv(k_i)+1)z_i]d\tau + \frac{1}{2}e_i^2(0)\\ &= \int_0^t [-e_i^2 - (\theta-\hat{\theta}_i)\xi_i e_i + (-bv(k_i)+1)z_i e_i]d\tau) + \frac{1}{2}e_i^2(0)\\ &= \int_0^t [-e_i^2 - (\theta-\hat{\theta}_i)\xi_i e_i + (-bv(k_i)+1)\dot{k}_i]d\tau + \frac{1}{2}e_i^2(0)\\ &= -\int_0^t e_i^2 d\tau - \int_0^t (\theta-\hat{\theta}_i)\xi_i e_i d\tau + \int_0^t (-bv(k_i)+1)\dot{k}_i d\tau + \frac{1}{2}e_i^2(0).\end{aligned}$$

Substituting the parameter updating law (4.9), and using the algebraic relationship $(a-b)^2-(a-c)^2 = -2(a-b)(b-c)-(b-c)^2$, the second term on the right hand side of (4.16) can be expressed as

$$\begin{aligned}&\frac{1}{2}\int_0^t(\phi_i^2-\phi_{i-1}^2)d\tau = \frac{1}{2}\int_0^t[(\theta-\hat{\theta}_i)^2-(\theta-\hat{\theta}_{i-1})^2]d\tau\\ &= -\int_0^t(\theta-\hat{\theta}_i)(\hat{\theta}_i-\hat{\theta}_{i-1})d\tau - \frac{1}{2}\int_0^t(\hat{\theta}_i-\hat{\theta}_{i-1})^2 d\tau\\ &= \int_0^t(\theta-\hat{\theta}_i)\xi_i e_i d\tau - \frac{1}{2}\int_0^t \xi_i^2 e_i^2 d\tau. \end{aligned} \tag{4.17}$$

Therefore, the difference of the composite energy function is

$$\begin{aligned}\Delta E_i(t) &= -\int_0^t e_i^2 d\tau - \frac{1}{2}\int_0^t \xi_i^2 e_i^2 d\tau + \int_0^t(-bv(k_i)+1)\dot{k}_i d\tau\\ &\quad + \frac{1}{2}e_i^2(0) - \frac{1}{2}e_{i-1}^2(t) + \frac{1}{2}\int_t^T(\phi_{i-1}^2-\phi_{i-2}^2)d\tau. \end{aligned} \tag{4.18}$$

Let $t=T$, according to Assumption 4.1 we have $\frac{1}{2}e_i^2(0) = \frac{1}{2}e_{i-1}^T(T)$. In the sequel

$$\begin{aligned}\Delta E_i(T) &= -\int_0^T e_i^2 d\tau - \frac{1}{2}\int_0^T \xi_i^2 e_i^2 d\tau + \int_0^T(-bv(k_i)+1)\dot{k}_i d\tau\\ &\le -\int_0^T e_i^2 d\tau + \int_0^T(-bv(k_i)+1)\dot{k}_i d\tau. \end{aligned} \tag{4.19}$$

Part II: Learning Convergence Property

Applying (4.19) repeatedly, we have

$$\begin{aligned} E_i(T) &= E_0(T) + \sum_{j=1}^{i} \Delta E_j(T) \\ &\leq E_0(T) - \sum_{j=1}^{i} \int_0^T e_j^2 d\tau + \sum_{j=1}^{i} \int_0^T (-bv(k_j)+1)\dot{k}_j d\tau. \end{aligned} \tag{4.20}$$

Define a new function $\dot{k}(t+(i-1)T) \triangleq \dot{k}_i(t)$, and $k(t+(i-1)T) = k_i(t)$ for $t \in [0, T]$. By virtue of Property 4.2 and the learning control law (4.8), $\dot{k}(t)$ is a continuous function and $k(t)$ is a C^1 function for $\forall t \in [0,\ iT]$. Thus

$$\begin{aligned} & \sum_{j=1}^{i} \int_0^T (-bv(k_j)+1)\dot{k}_j d\tau \\ = & \int_0^T (-bv(k_1)+1)\dot{k}_1 d\tau + \int_0^T (-bv(k_2)+1)\dot{k}_2 d\tau + \cdots + \int_0^T (-bv(k_i)+1)\dot{k}_i d\tau \\ = & \int_0^T (-bv(k)+1)\dot{k} d\tau + \int_T^{2T} (-bv(k)+1)\dot{k} d\tau + \cdots + \int_{(i-1)T}^{iT} (-bv(k)+1)\dot{k} d\tau \\ = & \int_0^{iT} (-bv(k)+1)\dot{k} d\tau. \end{aligned} \tag{4.21}$$

Denote $V(\tau+(i-1)T) = E_i(\tau)$, from (4.20) we have

$$V(iT) + \sum_{j=1}^{i} \int_0^T e_j^2 d\tau \ \leq \ E_0(T) + \sum_{j=1}^{i} \int_0^T (-bv(k_j)+1)\dot{k}_j d\tau.$$

Then

$$\begin{aligned} V(iT) &\leq E_0(T) + \sum_{j=1}^{i} \int_0^T (-bv(k_j)+1)\dot{k}_j d\tau - \sum_{j=1}^{i} \int_0^T e_j^2 d\tau \\ &= E_0(T) + \int_0^{iT} (-bv(k)+1)\dot{k} d\tau - \sum_{j=1}^{i} \int_0^T e_j^2 d\tau. \end{aligned} \tag{4.22}$$

Furthermore, the upper right hand derivative of $E_i(t)$ should be

$$\dot{E}_i(t) = e_i\dot{e}_i + \frac{1}{2}(\phi_i^2(t) - \phi_{i-1}^2(t))$$

Substituting the error dynamics in (4.13), the first term on the right hand side is

$$\begin{aligned} e_i\dot{e}_i &= e_i(-e_i - (\theta - \hat{\theta}_i)\xi_i + (-bv(k_i)+1)z_i) \\ &= -e_i^2 - (\theta - \hat{\theta}_i)\xi_i e_i + (-bv(k_i)+1)z_i e_i. \end{aligned} \tag{4.23}$$

Similarly as (4.17), we obtain

$$\frac{1}{2}(\phi_i^2(t) - \phi_{i-1}^2(t)) = (\theta - \hat{\theta}_i)\xi_i e_i - \frac{1}{2}\xi_i^2 e_i^2. \tag{4.24}$$

Therefore the upper right hand derivation of E_i is

$$\begin{aligned} \dot{E}_i(t) &= -e_i^2 + (-bv(k_i) + 1)z_i e_i - \frac{1}{2}\xi_i^2 e_i^2 \\ &\leq (-bv(k_i) + 1)\dot{k}_i. \end{aligned} \tag{4.25}$$

Thus based on (4.22), for $\forall t \in [0,\ T]$ we have

$$\begin{aligned} V(iT + t) &= V(iT) + \int_0^t \dot{E}_{i+1}(\tau)d\tau \\ &\leq E_0(T) - \sum_{j=1}^{i} \int_0^T e_j^2 d\tau + \int_0^{iT} (-bv(k) + 1)\dot{k}d\tau \\ &\quad + \int_0^t (-bv(k_{i+1}) + 1)\dot{k}_{i+1} d\tau \\ &\leq E_0(T) - \sum_{j=1}^{i} \int_0^T e_j^2 d\tau + \int_0^{iT} (-bv(k) + 1)\dot{k}d\tau \\ &\quad + \int_{iT}^{(iT+t)} (-bv(k) + 1)\dot{k}d\tau \\ &= E_0(T) - \sum_{j=1}^{i} \int_0^T e_j^2 d\tau + \int_0^{(iT+t)} (-bv(k) + 1)\dot{k}d\tau, \end{aligned}$$

i.e.,

$$\lim_{i\to\infty} V(iT + t) \leq E_0(T) - \lim_{i\to\infty} \sum_{j=1}^{i} \int_0^T e_j^2 d\tau + \lim_{i\to\infty} \int_0^{(iT+t)} (-bv(k) + 1)\dot{k}d\tau.$$

According Property 4.1,

$$\lim_{i\to\infty} \int_0^{(iT+t)} (-bv(k) + 1)\dot{k}d\tau \leq B, \tag{4.26}$$

where B is a finite positive constant. In the sequel we can derive

$$\lim_{i\to\infty} V(iT + t) \leq E_0(T) + B - \lim_{i\to\infty} \sum_{j=1}^{i} \int_0^T e_j^2 d\tau. \tag{4.27}$$

If $E_0(T)$ is a finite number, considering the positiveness of $V(iT + t)$, and boundedness of B, (4.27) implies $e_i(t) \to 0$ in L_T^2 as $i \to \infty$.

Part III: the Finiteness of $E_0(T)$

Now we prove the finiteness of $E_0(t)$ $\forall t \in [0,\ T]$. The finiteness property is necessary, as $\xi(x,t)$ may be a local Lipschitz continuous function and finite escape time phenomenon may occur. From the system dynamics (4.1) and the proposed control laws (4.8) and (4.9), it can be derived that the right hand side of (4.1) is continuous with respect to all the arguments. According to the existence theorem of differential equation (Yoshizawa, 1966), there exists a solution in an interval $[0,\ T_1) \subset [0,\ T]$, where $T_1 > 0$. Therefore, the boundedness of $E_0(t)$ over $[0,\ T_1]$ can be guaranteed and we need only focus on the interval $(T_1,\ T]$.

For any $t \in (T_1,\ T]$, the derivative of $E_0(t)$ is

$$\dot{E}_0 = e_0\dot{e}_0 + \frac{1}{2}\phi_0^2. \tag{4.28}$$

At the first iteration $i = 0$, $\hat{\theta}_{-1}(t) = 0$, thus

$$\hat{\theta}_0 = -\gamma_0(t)\xi_0 e_0.$$

Since $\gamma_0(t)$ is strictly increasing in $[0,\ T]$, $\frac{1}{\gamma_0(t)} \geq 1$ is ensured in the time interval $(T_1,\ T]$. Substituting (4.13) and the parameter updating law (4.9) into $\dot{E}_0$ yields

$$\begin{aligned}
\dot{E}_0 &= e_0\dot{e}_0 + \frac{1}{2}(\hat{\theta}_0 - \theta)^2 \\
&\leq e_0\dot{e}_0 + \frac{1}{2\gamma_0(t)}(\hat{\theta}_0 - \theta)^2 \\
&= e_0[-e_0 - (\theta - \hat{\theta}_0)\xi_0 + (-bv(k_0) + 1)z_0] + (\theta - \hat{\theta}_0)\xi_0 e_0 \\
&\quad -\frac{1}{2\gamma_0(t)}\hat{\theta}_0^2 + \frac{1}{2\gamma_0(t)}\theta^2 \\
&= -e_0^2 + (-bv(k_0) + 1)z_0 e_0 - \frac{1}{2\gamma_0(t)}\hat{\theta}_0^2 + \frac{1}{2\gamma_0(t)}\theta^2 \\
&= -e_0^2 + (-bv(k_0) + 1)\dot{k}_0 - \frac{1}{2\gamma_0(t)}\hat{\theta}_0^2 + \frac{1}{2\gamma_0(t)}\theta^2.
\end{aligned} \tag{4.29}$$

Integrating both sides of the above inequality from T_1 to t we have

$$\begin{aligned} E_0(t) &= E_0(T_1) - \int_{T_1}^t e_0^2 d\tau + \int_{T_1}^t (-bv(k_0)+1)\dot{k}_0 d\tau \\ &\quad - \int_{T_1}^t \frac{\hat{\theta}_0^2}{2\gamma_0(\tau)} d\tau + \int_{T_1}^t \frac{\theta^2}{2\gamma_0(\tau)} d\tau \\ &\leq E_0(T_1) + \int_{T_1}^t (-bv(k_0)+1)\dot{k}_0 d\tau + \int_{T_1}^t \frac{\theta^2}{2\gamma_0(\tau)} d\tau. \end{aligned} \tag{4.30}$$

Since $\theta(t) \in C[0,\ T]$, $\int_{T_1}^t \frac{\theta^2}{2\gamma_0(\tau)} d\tau$ is bounded. Finally applying Property 4.1 to (4.30), we can conclude both $\int_{T_1}^t (-bv(k_0)+1)\dot{k}_0 d\tau$ and $E_0(t)$ are finite over $(T_1,\ T]$. Thus $E_0(t)$ is bounded on $[0,\ T]$. □

Remark 4.1. *The above results can be extended straightforward to the system*

$$\dot{x} = \boldsymbol{\theta}(t)\boldsymbol{\xi}(x,t) + bu, \qquad x(0) = x_0, \tag{4.31}$$

where $\boldsymbol{\theta}(t) = [\theta_1(t), \theta_2(t), \cdots, \theta_n(t)]$ *and* $\boldsymbol{\xi}(x) = [\xi_1(x), \xi_2(x), \cdots, \xi_n(x)]^T$.

Accordingly we should replaced $\hat{\theta}_i$ *by* $\hat{\boldsymbol{\theta}}_i$ *and* ξ_i *by* $\boldsymbol{\xi}_i$ *in the learning mechanism, and replace* ϕ_i^2 *in CEF by* $\boldsymbol{\phi}_i^T\boldsymbol{\phi}_i$ *with* $\boldsymbol{\phi}_i = \boldsymbol{\theta} - \hat{\boldsymbol{\theta}}_i$.

Remark 4.2. *To improve the learning control performance, we can add a positive gain* γ *to both differential and difference updating laws, such that* $\dot{k}_i = \gamma z_i e_i$ *and*

$$\hat{\theta}_i(t) = \begin{cases} 0, & i = -1, \\ -\gamma_0(t)\xi(x_i)e_i(t), & i = 0, \\ \hat{\theta}_{i-1}(t) - \gamma\xi(x_i)e_i(t), & i \geq 1, \end{cases} \tag{4.32}$$

where $\gamma_0(t)$ *is defined analogously as before except that* $\gamma_0(T) = \gamma$. *The convergence analysis remains the same except for the CEF which should be changed to*

$$E_i(t) = \frac{1}{2\gamma}e_i(t)^2 + \frac{1}{2\gamma}\int_0^t \phi_i^2(\tau)d\tau + \frac{1}{2\gamma}\int_t^T \phi_{i-1}^2(\tau)d\tau.$$

4.4 An Illustrative Example

Consider the system (4.1), where $\xi(x) = x^2$, $\theta(t) = 1 + sin\pi t$, and $b = 1$ which is assumed unknown. The reference model is

$$\dot{x}_r = -cos\pi t\ x_r - 2cos\pi t.$$

Let $t \in [0,\ 2]$, $x_0(0) = 1$ and $x_r(0) = 0$. Applying the learning control (4.8), the simulation result is shown in Figure 4.1. The horizontal axis denotes the number of iterations, and the vertical axis denotes the sup-norm $|e_i|_{sup}$, i.e., the maximum tracking error of $|e_i(t)|$ over $[0,\ 2]$. The learning convergence can be clearly seen. Figure 4.2 shows the evolution of the Nussbaum-type function $v(k_i(t))$ over the

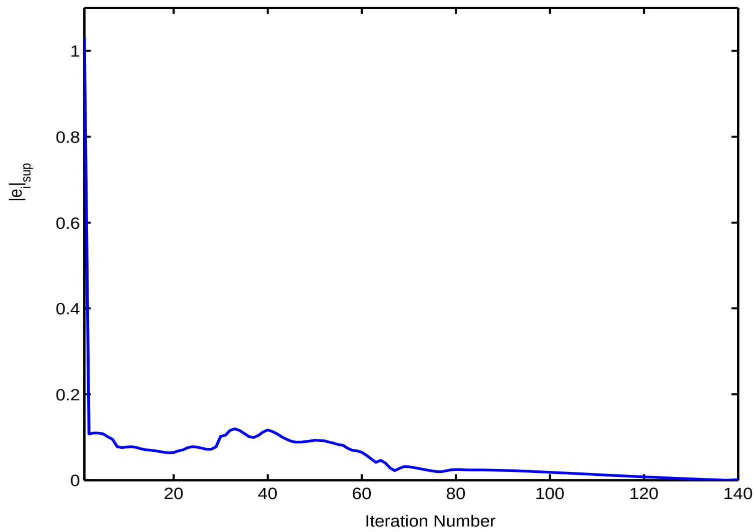

Figure 4.1. Learning convergence of ILC based on CEF, $t \in [0,\ 2]$.

iterations, where the dashed line and solid line denote respectively the lower and upper bounds of $v(k_i(t))$ at each iteration. It finally converges to a positive value, hence is consistent with the actual sign of the system parameter $b = 1$. On the other hand, we can also observe the swing phenomenon between “+” and “-”, which reflects the transient behavior of the adaptation process. Nevertheless, the iterative learning retains a fast convergence.

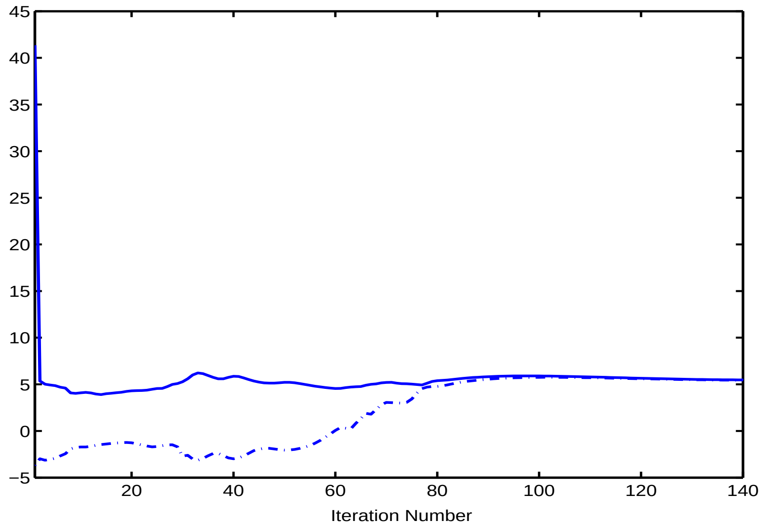

Figure 4.2. Evolution of the Nussbaum gain $v(\cdot)$.

4.5 Conclusion

To deal with the tracking problem without *a priori* knowledge of the control direction, we incorporate the Nussbaum-type function into the learning control design. Based on the idea of composite energy function, the proposed learning control mechanism achieves the L_T^2 convergence of the tracking error sequence in the iteration domain. The effectiveness of the ILC design is demonstrated through a numerical example.

Chapter 5

Adaptive Learning Control for Finite Interval Tracking Based on Constructive Function Approximation and Wavelet

5.1 Introduction

Learning control (Arimoto *et al.*, 1984*a*), (Lee and Bien, 1997), (Moore, 1998), (Sun and Wang, 2001) or adaptive learning control (ALC) (Xu and Badrinath, 2000) and (French and Rogers, 2000*a*), developed as the complementary to adaptive control, can cope with any tracking control tasks repeated over a finite time interval. Unlike adaptive control that targets at asymptotic convergence along the time axis, learning control targets at perfect tracking over a finite interval by means of asymptotic convergence along the learning axis (iteration axis). In this chapter, we focus on adaptive learning control with the ultimate objective of addressing the finite interval tracking problems.

A constantly challenging mission for control society is to deal with dynamic systems in the presence of unknown nonlinearities. Consider the following simple affine dynamics

$$\dot{x} = f(x) + u$$

where u is the system input. Over the past five decades, numerous control strategies have been developed according to the characteristic and prior knowledge of $f(x)$. If $f(x)$ can be parameterized as the product of unknown time invariant parameters and known nonlinear functions, adaptive control and adaptive learning are most suitable. If $f(x)$ cannot be parameterized but its upperbounding function $\bar{f}(x)$ is known *a priori*, robust control or robust learning control (Tan and Xu, 2003) is pertinent. In the past decade, intelligent control methods using function approximation, such as neural network, fuzzy network, and wavelet network, have been proposed, which open a new avenue leading to more generic solutions and better control performance. The most profound feature of those function approximation methods lies in that the non-parametric function $f(x)$ is given a representation in a parameter space. Hence the control problem renders into an analogy as the adaptive control or adaptive learning control: only dealing with unknown time invariant parameters.

Neural network based control is most widely studied (Narendra and Parthasarathy, 1990), (Hunt *et al.*, 1992), (Levin and Narendra, 1996), (Sanner and Slotine, 1992), (Polycarpou, 1996), (Seshagiri and Khalil, 2000), (Ge and Wang, 2002) and (Huang *et al.*, 2003). The success of neural control is subject to the validity of a prerequisite: the structure of the network, such as the number of layers and nodes, must be adequate to meet the desired approximation precision. Hence, it is commonly assumed in adaptive neural control, that for a continuous function $f(x)$ on a compact set, a finite and sufficiently large neural network is chosen and there exists a set of ideal weights $\boldsymbol{\theta}$ such that the function can be approximated to a specified precision (Poggio and Girosi, 1990). It was indicated in (Gupta and Rao, 1994), (Funahashi, 1989) and (Hornik *et al.*, 1989) that if the node number of a three layer neural network is adequate, the approximation error can be arbitrarily small on a compact set.

Due to the lack of prior information on $f(x)$, often a designer is unable to know how large a neural network would be adequate. If the network structure is inadequate, the control mission is impossible. Intuitively, a solution to this problem is to let the neural network evolves continuously from a small initial configuration and ceases only when the desired precision is satisfied. However we encounter a difficulty when implementing this idea with adaptive neural control, because a neural network is constructed as a complete system instead of a basis. The fundamental difference between a complete system and a basis can be clearly seen from the changes of weights when the system structure evolves (Lebedev *et al.*, 1994). The new weights of a complete system, $\boldsymbol{\theta}_A$, may be totally different from the original weights, $\boldsymbol{\theta}$. On the other hand, the new weights of a basis, $\boldsymbol{\theta}_A$, will include the original weights, $\boldsymbol{\theta}$, as an invariant subset. Hence, after adding new nodes to a neural network, parametric adaptation may have to restart from scratch for the new weights $\boldsymbol{\theta}_A$. Using a basis in approximation, on the other hand, the adaptively learned results for weights $\boldsymbol{\theta}$ will remain valid and thus adaptive learning can be carried on. Adaptive learning will start from beginning only for newly added weights in $\boldsymbol{\theta}_A$.

In this chapter, we consider two scenarios. In the first scenario, $f(x)$ is assumed global $\mathcal{L}^2$, i.e. $\mathcal{L}^2(R)$, which is the only prior knowledge. ALC can generate a convergent sequence and enter the pre-specified bound in a finite number of learning iterations. In the second scenario, $f(x)$ is assumed local $\mathcal{L}^2$, and the prior knowledge is the upperbound $\bar{f}(x)$. A robust control mechanism is applied first to confine the state x to a compact set. By augmenting $f(x)$ to a new function defined on R, we show that the second scenario renders to the first one, consequently achieves the same convergence property with ALC. With the help of Lyapunov method, a rigorous analysis is conducted in order to disclose the inherent properties of the proposed adaptive learning control system, including the existence of the solution, the asymptotic convergence along the learning axis, and the tracking performance with the designated error bound. Extension to more general plants, either with a partially unknown input coefficient, or in cascade form, will also be exploited.

Wavelet network, consisting of bases, has been developed as a universal function approximator in $\mathcal{L}^2$, thus its structure can easily evolve in conjunction with parametric adaptation or adaptive learning. In this chapter, three different wavelets are presented and their suitability are exploited. Through illustrative examples, we also demonstrate the relationship between the complexity of wavelet network and the number of learning iterations.

The chapter is organized as follows. In Section 5.2, the problem formulation and preliminaries are briefed. In Section 5.3, the adaptive learning control with universal function approximation is proposed. In Section 5.4, a robust adaptive learning control is proposed for local $\mathcal{L}^2$ nonlinear plants. In Section 5.5, ALC is applied to more generic nonlinear plants. In Section 5.6, the properties of wavelet approximation is presented. In Section 5.7, illustrative examples and design considerations are provided. In Section 5.8 the conclusion is given.

In the chapter we define

$\|\cdot\|$	a vector norm
$\|\cdot\|_2$	$\mathcal{L}^2$ − norm
$\|\cdot\|_s$	uniform norm
$\|\cdot\|_T$	extended $\mathcal{L}^2$ − norm,
	defined as $\|\cdot\|_T \triangleq \frac{1}{T}\int_0^T \|\cdot\|^2 d\tau$
$\|\mathbf{z}_i\|_m$	$\max\{\|z_{j,i}\|_s : j = 1, ..., n+i\}$
	for $\mathbf{z}_i = (z_{1,i}, ..., z_{n+i,i})^T$

In subsequent context, we omit the argument t for all variables where no confusion arises.

5.2 Problem Formulation and Preliminaries

First define a basis.

Definition 5.1. *Let Y be a normed linear space over real number field R. A system of elements $g_1, g_2, \cdots \subset Y$ is said to be a basis for Y if any element $y \in Y$ has a unique representation*

$$y = \sum_{k=1}^{\infty} \theta_k g_k, \tag{5.1}$$

with scalars $\theta_k \in R$.

Note that the meaning of (5.1) is: if $y_i = \sum_{k=1}^{i} \theta_k g_k$, then $\lim_{i\to\infty} \|y - y_i\| = 0$, where $\|\cdot\|$ is the norm in the space Y. For arbitrary $\epsilon > 0$, to make $\|y - y_i\| \leq \epsilon$ we simply take i large enough. Further, coefficients θ_1, θ_2, $\cdots$ are unique.

The existence and construction of a basis for a particular normed linear space could be very difficult in general. However it is well known that there exist orthonormal bases in Hilbert space. In particular there exist orthonormal wavelet bases in $\mathcal{L}^2(R)$.

To facilitate the subsequent discussions on the existence of solution, the following Lemma is introduced.

Lemma 5.1. *((Zheng* et al.*, 1991)) Consider the following Cauchy problem*

$$\dot{\mathbf{x}} = \mathbf{f}(t, \mathbf{x}), \qquad \mathbf{x}(t_0) = \mathbf{x}_0. \tag{5.2}$$

If $\mathcal{D}$ is an open set in R^{n+1}, $\mathbf{f} : \mathcal{D} \to R^n$ is continuous in $\mathcal{D}$ and satisfies locally Lipschitzian condition for $\mathbf{x}$, then the solution of Cauchy problem (7.7) can be extended to the boundary of $\mathcal{D}$ – $\partial\mathcal{D}$ ($\partial\mathcal{D}$ can be ∞).

To focus on the essential idea and properties of the proposed adaptive learning control, the following simple dynamic plant is considered first

$$S_I : \begin{cases} \dot{x}_j = x_{j+1}, & j = 1, 2, \cdots, n-1, \\ \dot{x}_n = f(\mathbf{x}) + u & \mathbf{x}(0) = \mathbf{x}_0, \end{cases} \tag{5.3}$$

where $\mathbf{x} = [x_1, x_2, \cdots, x_n]^T \in R^n$ is the state vector, and $u \in R$ is the plant input. The mapping $f(\mathbf{x})$ is an unknown nonlinear function which is continuous and locally Lipschitzian for $\mathbf{x} \in R^n$. We consider two types of prior knowledge of f that lead to two distinct ALC designs.

Assumption 5.1. $f(\mathbf{x}) \in \mathcal{L}^2(R^n)$.

A ALC method is developed for S_I satisfying assumption 5.1.

Assumption 5.2. *$f(\mathbf{x}) \in \mathcal{L}^2(\mathcal{D})$ where $\mathcal{D} \in R^n$ is a compact set. There exists a known continuous function $\bar{f}(\mathbf{x}) \geq 0$ such that $|f(\mathbf{x})| \leq \bar{f}(\mathbf{x}), \forall \mathbf{x} \in \mathcal{D}$.*

For S_I satisfying assumption 5.2, a robust ALC is proposed in this chapter.

ALC is further extended to two classes of more general plants. One class is described by

$$S_{II}: \begin{cases} \dot{x}_j &= x_{j+1}, \qquad\qquad j = 1, 2, \cdots, n-1, \\ \dot{x}_n &= f(\mathbf{x}) + b(t, \mathbf{x})u \quad \mathbf{x}(0) = \mathbf{x}_0, \end{cases} \tag{5.4}$$

where f has a bounding function $\bar{f}$, and $b(t, \mathbf{x})$ is a partially unknown function satisfying the following condition.

Assumption 5.3. $b(\mathbf{x}) \geq b_0 > 0, \quad \forall\, \mathbf{x} \in R^n$.

The other class is the n-th order cascade dynamics

$$S_{III}: \begin{cases} \dot{x}_j &= f_j(\mathbf{x}_j) + x_{j+1}, \\ \dot{x}_n &= f_n(\mathbf{x}) + u, \end{cases} \tag{5.5}$$

where $\mathbf{x}_j = [x_1, \cdots, x_j]^T$, and $f_j(\mathbf{x}_j) \in \mathcal{L}^2(R^j)$ are nonlinear unknown functions. It is known that f_j $(j = 1, \cdots, n-1)$ are unmatched uncertainties.

Now give the control objective. Let $x_r(t) \in \mathcal{C}^n[0, T')$ be a n-th order continuously differentiable trajectory, then $x_r, x_r^{(1)}, \cdots, x_r^{(n)}$ are bounded on a finite interval $[0,\ T]$, where $T' > T$. Define $\mathbf{x}_r \triangleq [x_r, x_r^{(1)}, \cdots, x_r^{(n-1)}]^T$ and $\Delta\mathbf{x}_i = \mathbf{x}_i - \mathbf{x}_r = [\Delta x_{1,i}, \Delta x_{2,i}, \cdots, \Delta x_{n,i}]^T$, where $\mathbf{x}_i = [x_{1,i}, x_{2,i}, \cdots, x_{n,i}]$ is the state vector at the $i-$th learning iteration. An augmented tracking error σ_i at the i-th learning iteration is defined as

$$\sigma_i = (\frac{d}{dt} + \lambda)^{n-1} \Delta x_{1,i} = [\boldsymbol{\lambda}^T\ 1]\Delta\mathbf{x}_i, \tag{5.6}$$

where $\boldsymbol{\lambda} = [\lambda^{n-1}, (n-1)\lambda^{n-2}, \cdots, (n-1)\lambda]^T$ with $\lambda > 0$.

The ultimate control objective is to find a sequence of appropriate control input, $u_i(t)$, $t \in [0,\ T]$, such that the tracking error sequence will enter a pre-specified bound in $\mathcal{L}_T^2$, after a finite number of learning iterations. Here the tracking error sequence is the augmented one, σ_i, for plants S_I and S_{II}, and $x_{1,i} - x_r$ for the plant S_{III}.

5.3 Adaptive Learning Control

In this section, a new adaptive learning control approach based on function approximation is presented for the plant S_I in (5.3), whereby $f(\mathbf{x})$ meets Assumption 5.1.

Suppose that $g_1(\mathbf{x})$, $g_2(\mathbf{x})$, $\cdots$ form a continuous and locally Lipschitzian basis in the space $\mathcal{L}^2(R^n)$, then

$$f(\mathbf{x}) = \sum_{k=1}^{\infty} \theta_k g_k(\mathbf{x}), \tag{5.7}$$

with θ_k being unknown weights. Denote the approximation error

$$e_i(\mathbf{x}) = f(\mathbf{x}) - \sum_{k=1}^{i} \theta_k g_k(\mathbf{x}). \tag{5.8}$$

It is obvious that

$$\lim_{i\to\infty} \|e_i\|_T = \lim_{i\to\infty} \int_{\mathcal{R}^n} \|e_i\|_2 d\mathbf{x} = 0. \tag{5.9}$$

If the basis is sufficiently smooth and well localized, then the series expansion of continuous square integrable functions in fact also converges pointwisely. For example, if we choose wavelet as a basis, then the convergence of the resulting series in an $\mathcal{L}^2$ sense should also be in pointwise sense under appropriate constraints on the wavelet (Kelly *et al.*, 1994) and (Walter, 1995). These additional smoothness and decay conditions on the basis are assumed throughout the analysis in this chapter. Note that the pointwise convergence of $e_i(\mathbf{x})$ holds $\forall \mathbf{x} \in \mathcal{R}^n$. Suppose $\mathbf{x}$ is a vector valued function of the time t, and let $t \in [0,\ T]$, then $\mathbf{x}(t)$ is a map $\mathbf{x} : [0,\ T] \to \mathcal{D} \subset \mathcal{R}^n$. Obviously $e_i(\mathbf{x}(t))$ is pointwise over $\mathcal{D}$, thus $e_i(\mathbf{x}(t))$ is a compound function pointwise convergent in $[0,\ T]$, in the sequel $\|e_i\|_T$ is a convergent sequence, namely

$$\lim_{i\to\infty} \|e_i\|_T = \lim_{i\to\infty} \int_0^T |e_i(t)|^2 dt = 0. \tag{5.10}$$

From the above convergence property, there exists a constant M such that $\|e_i\|_T \leq M$ for any i.

Since the learning control objective is to track a given trajectory in a finite interval, it is well know that the initial state values will directly affect the learning results (Xu and Yan, 2005). In this chapter, we consider 5 types of initial conditions from the practical point of view

Assumption 5.4.

a) $\sigma_i(0) = 0$*;*

b) $\sum_{i=1}^{\infty} \sigma_i^2(0) = \sigma_0$*, where* σ_0 *is a constant;*

c) $|\sigma_i(0)| = \sigma_0 \neq 0$*, where* σ_0 *is a constant;*

d) $\sigma_i(0)$ *is random and bounded by a constant* σ_0*;*

e) $\sigma_i(0) = \sigma_{i-1}(T)$*, and* $|\sigma_1(0)| \leq \sigma_0$*.*

Condition a) is the typical identical initialization condition; condition b) implies that $\sigma_i(0)$ belongs to l^2; condition c) is the fixed initial shift; condition d) includes first three conditions as the special cases; and condition e) is the alignment condition often seen in processes without a resetting mechanism (Xu and Yan, 2005).

Consider system S_I in (5.3), the tracking error dynamics at the i-th learning iteration can be expressed as

$$\dot{\sigma}_i = f(\mathbf{x}_i) + u_i(t) + v(t, \mathbf{x}_i), \quad \forall t \in [0,\ T] \tag{5.11}$$

where $v(t, \mathbf{x}_i) \triangleq -x_r^{(n)}(t) + [0\ \boldsymbol{\lambda}]\Delta\mathbf{x}_i$.

For notational convenience, in the following $f(\mathbf{x}_i)$, $g_k(\mathbf{x}_i)$ and $v_i(t, \mathbf{x}_i)$ are denoted by f_i, g_i and v_i respectively.

The adaptive learning control mechanism is given as

$$u_i = -\beta\sigma_i - \hat{\boldsymbol{\theta}}_i^T \mathbf{g}_i - v_i, \tag{5.12}$$

where $\hat{\boldsymbol{\theta}}_i = [\hat{\theta}_1, \cdots, \hat{\theta}_{k(i)}]^T$ and $\mathbf{g}_i = [g_1, \cdots, g_{k(i)}]^T$ with $k = k(i)$. $k(i)$ is a function of the number of iterations i, reflecting how frequently a new base is added to the existing basis set. For instance, one can add a new base g_k to the existing set $g_1, \cdots, g_{k-1}$ after every 10 learning iterations. A possible relationship between k and i is given in the Figure 5.1. For simplicity, let $k(i) = i$ in the theory proof. This implies that the function approximation network is updated at every learning iteration. The parametric adaptive learning law is

$$\begin{aligned} \dot{\hat{\boldsymbol{\theta}}}_i &= \sigma_i \mathbf{g}_i, \\ \hat{\boldsymbol{\theta}}_1(0) &= 0, \quad \hat{\boldsymbol{\theta}}_i(0) = \hat{\boldsymbol{\theta}}_{i-1}(T). \end{aligned} \tag{5.13}$$

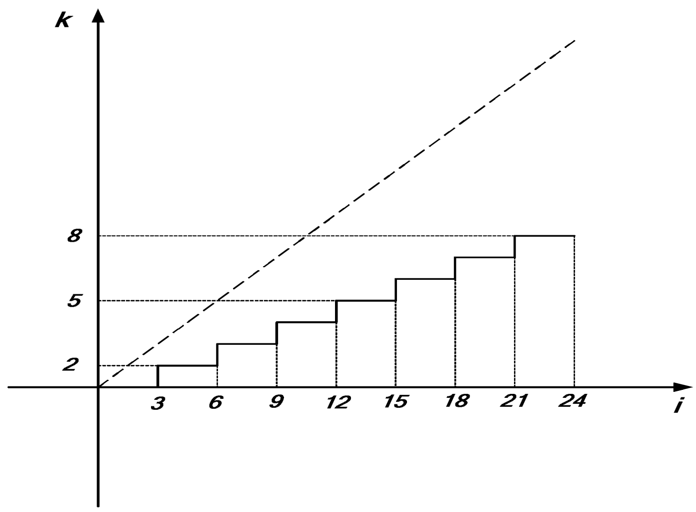

Figure 5.1. Update the structure for every 3 iterations

Substituting the adaptive learning control law (5.12) into the tracking error dynamics (5.11) yields

$$\begin{aligned} \dot{\sigma}_i &= f_i + u_i + v_i \\ &= -\beta\sigma_i + (\boldsymbol{\theta}_i - \hat{\boldsymbol{\theta}}_i)^T \mathbf{g}_i + e_i. \end{aligned} \tag{5.14}$$

Define the augmented state vector $\mathbf{z}_i \triangleq (\mathbf{x}_i, \hat{\boldsymbol{\theta}}_i)$. From the plant (5.3), adaptive learning mechanism (5.13), and ALC sequence (5.12), we have

$$\dot{\mathbf{z}}_i = \mathbf{h}(t, \mathbf{z}_i), \tag{5.15}$$

where

$$\begin{aligned} \mathbf{h}(t, \mathbf{z}_i) &= [x_{2,i}, \cdots, x_{n,i}, h_{\mathbf{x}}(t, \mathbf{z}_i), \mathbf{h}_{\hat{\theta}}^T(t, \mathbf{z}_i)]^T, \\ h_{\mathbf{x}}(t, \mathbf{z}_i) &= f_i + u_i \\ &= -\beta[\boldsymbol{\lambda}\ 1]\mathbf{x}_i - v_i + \boldsymbol{\theta}_i^T \mathbf{g}_i - \hat{\boldsymbol{\theta}}_i^T \mathbf{g}_i + e_i + \beta[\boldsymbol{\lambda}\ 1]\mathbf{x}_r, \\ \mathbf{h}_{\hat{\theta}}(t, \mathbf{z}_i) &= [\boldsymbol{\lambda}^T\ 1]\Delta\mathbf{x}_i \mathbf{g}_i. \end{aligned} \tag{5.16}$$

The first main result, which is concerned with the existence of solution of the above augmented dynamics (5.15) under the initial conditions described in Assumption 5.4, is summarized in the following theorem.

Theorem 5.1. *The solution $\mathbf{z}_i$ exists in $[0,\ T]$ by choosing the feedback gain $\beta > 1$.*

Proof. Since the control task ends in the finite interval $[0,\ T]$, all we need to prove is no finite escape time for $\mathbf{z}_i$ in $[0,\ T]$. We shall prove that the solution $\mathbf{z}_i(t)$ of the dynamic system (5.15) exists in $[0,\ T')$, which therefore implies the existence in $[0,\ T]$. Define $\Omega \triangleq R^{n+i} \times [0, T')$

Clearly, $\mathbf{h}(t, \mathbf{z}_i) : \Omega_i \to R^{n+i}$ is continuous. By Peano's Existence Theorem (Zheng *et al.*, 1991), associated with the initial values $\mathbf{z}_i(0) = (\mathbf{x}_0, \hat{\boldsymbol{\theta}}_i(0)) \in \Omega_i$, equation (5.15) has a continuous solution in a neighborhood of $t = 0$. Furthermore it is easy to check that $\mathbf{h}(t, \mathbf{z}_i)$ is locally Lipschitz continuous in $\mathbf{z}_i$. We only need to consider the solution for $t > 0$. Let $[0,\ t_i)$ be the maximal interval to which the solution $\mathbf{z}_i(t)$ can be continued up. Lemma 5.1 implies that $\mathbf{z}_i(t)$ tends to the boundary $\partial\Omega_i$ as $t \to t_i$. It further implies that $\lim_{t \to t_i} \|\mathbf{z}_i(t)\|_m = \infty$ if $t_i < T'$, i.e., for any $C > 0$ and for each i, there exists $\delta_i > 0$ such that $\|\mathbf{z}_i(t)\|_m \geq C$ for all $t \geq t_i - \delta_i$. Since $\mathbf{z}_i(t)$ exists for all $t \in [0, t_i - \frac{\delta_i}{2}]$, define a Lyapunov function

$$V(\sigma_i, \tilde{\boldsymbol{\theta}}_i) = \frac{1}{2}\sigma_i^2 + \frac{1}{2}\tilde{\boldsymbol{\theta}}_i^T \tilde{\boldsymbol{\theta}}_i, \tag{5.17}$$

where $\tilde{\boldsymbol{\theta}}_i = \boldsymbol{\theta}_i - \hat{\boldsymbol{\theta}}_i$. Differentiating $V(\sigma_i, \tilde{\boldsymbol{\theta}}_i)$ with respect to time t yields

$$\dot{V}(\sigma_i, \tilde{\boldsymbol{\theta}}_i) = \sigma_i \dot{\sigma}_i - \tilde{\boldsymbol{\theta}}_i^T \dot{\hat{\boldsymbol{\theta}}}_i. \tag{5.18}$$

Substituting the augmented error dynamics (5.14) and the parametric adaptive learning law (5.13) yields

$$\dot{V}(\sigma_i, \tilde{\boldsymbol{\theta}}_i) = -\beta \sigma_i^2 + \sigma_i e_i. \tag{5.19}$$

Using Young's inequality, there exists $c \in (0,\ 1)$ such that

$$\sigma_i e_i \leq c\sigma_i^2 + \frac{1}{4c} e_i^2. \tag{5.20}$$

It follows from (5.19) that

$$\dot{V}(\sigma_i, \tilde{\boldsymbol{\theta}}_i) \ \leq \ (c - \beta)\sigma_i^2 + \frac{1}{4c} e_i^2 \tag{5.21}$$

where $c - \beta < 0$.

Next we will complete the proof by the mathematical induction. For $i = 1$, from Assumption 5.4, $|\sigma_1(0)| \leq \sigma_0$ for all initial conditions, and $\hat{\boldsymbol{\theta}}_1(0) = 0$. It follows from (5.21) and $\|e_i\|_T \leq M$ that

$$\begin{aligned} 0 \leq V(\sigma_1, \tilde{\boldsymbol{\theta}}_1) &= \int_0^t \dot{V}(\sigma_1, \tilde{\boldsymbol{\theta}}_1) d\tau + V(0,0) \\ &\leq \frac{M}{4c} + \frac{1}{2}\sigma_0^2 + \frac{1}{2}\boldsymbol{\theta}_1^2 \triangleq \frac{M_1^2}{4} \end{aligned}$$

for all $t \in [0, t_1 - \frac{\delta_1}{2}]$, i.e, $V(\sigma_1, \hat{\boldsymbol{\theta}}_1)$ is bounded on $[0, t_1 - \frac{\delta_1}{2}]$ by a constant which does not depend on δ_1. By the definition of Lyapunov function V, it can be derived from the above relationship that $|\sigma_1|_s \leq M_1$ and $|\hat{\theta}_1| \leq M_1$. Therefore, $\|\mathbf{z}_1(t)\|_m \leq M_1$ for all $t \in [0, t_1 - \frac{\delta_1}{2}]$. Note $M_1 > 0$ is a constant independent of δ_1. Taking $C = 2M_1$ in advance, for the corresponding $\delta_1 > 0$ we have

$$C \leq \|\mathbf{z}_1(t_1 - \frac{\delta_1}{2})\|_m \leq M_1 = \frac{C}{2}, \tag{5.22}$$

a contradiction which implies $t_1 \geq T'$.

Assume that $t_j \geq T'$ for $j = 2, \cdots, i-1$. Then the solution $\mathbf{z}_j(t)$ exists in $[0,\ T')$ and therefore σ_j and $\hat{\boldsymbol{\theta}}_j$ are both bounded for all $t \in [0,\ T]$. If $t_i < T'$, we have $\|\mathbf{z}_i(t)\|_m \geq C$ for all $t \geq t_i - \delta_i$, as shown above. Note that $|\sigma_i(0)| \leq \sigma_0$ for initial conditions a-d), $\sigma_i(0) = \sigma_{i-1}(T)$ for the initial condition e), and $\hat{\boldsymbol{\theta}}_i(0) = \hat{\boldsymbol{\theta}}_{i-1}(T)$. Hence quantities $\sigma_i(0)$ and $\hat{\boldsymbol{\theta}}_i(0)$ are bounded by a constant independent of δ_i. From (5.21) and $\mathcal{L}_T^2$ convergence property of e_i, we have

$$
\begin{aligned}
0 \leq V(\sigma_i, \tilde{\boldsymbol{\theta}}_i) &= \int_0^t \dot{V}(\sigma_i, \tilde{\boldsymbol{\theta}}_i)\, d\tau + V(\sigma_i(0), \tilde{\boldsymbol{\theta}}_i(0)) \\
&\leq \frac{M}{4c} + \frac{1}{2}\sigma_i(0)^2 + \frac{1}{2}(\boldsymbol{\theta}_i - \hat{\boldsymbol{\theta}}_i(0))^T(\boldsymbol{\theta}_i - \hat{\boldsymbol{\theta}}_i(0)) \triangleq \frac{M_i^2}{4} \quad (5.23)
\end{aligned}
$$

for all $t \in [0, t_i - \frac{\delta_i}{2}]$, i.e, $V(\sigma_i, \tilde{\boldsymbol{\theta}}_i)$ is bounded on $[0, t_i - \frac{\delta_i}{2}]$ by a constant which does not depend on δ_i. The definition of Lyapunov function V also implies that $\|\mathbf{z}_i(t)\|_m \leq M_i$ for all $t \in [0, t_i - \frac{\delta_i}{2}]$. By taking $C = 2M_i$, it leads to a contradiction analogous to (5.22). As a result, $t_i \geq T'$. □

For the closed-loop dynamic system (5.14) with the parametric updating law (5.13), the convergence property associated with initial conditions in Assumption 5.4 is displayed in the following theorem.

Theorem 5.2.

Part 1) Under the initial conditions a), b) and e), the exists a subsequence, $\{\sigma_{i_j}\}$ of $\{\sigma_i\}$, which enters any pre-specified bound ϵ after a finite number of learning iterations.

Part 2) Under the initial condition c) and d), for any arbitrary $\delta > 0$ and a bound given by $\epsilon = \frac{\sigma_0^2+\delta}{(\beta-c)T}$, there exists a subsequence, $\{\sigma_{i_j}\}$ of $\{\sigma_i\}$, which enters the given bound ϵ after a finite number of learning iterations.

Proof. Integrating both sides of (5.21) from 0 to T, and use the fact $\tilde{\boldsymbol{\theta}}_i(0) =$

$\tilde{\boldsymbol{\theta}}_{i-1}(T)$,

$$
\begin{aligned}
V(\sigma_i(T), \tilde{\boldsymbol{\theta}}_i(T)) &= V(\sigma_i(0), \tilde{\boldsymbol{\theta}}_i(0)) + \int_0^T \dot{V} dt \\
&\le V(\sigma_{i-1}(T), \tilde{\boldsymbol{\theta}}_{i-1}(T)) + V(\sigma_i(0), \tilde{\boldsymbol{\theta}}_{i-1}(T)) - V(\sigma_{i-1}(T), \tilde{\boldsymbol{\theta}}_{i-1}(T)) \\
&\quad -(\beta - c) \int_0^T \sigma_i^2 dt + \frac{1}{4c} \int_0^T e_i^2 dt \\
&= V(\sigma_{i-1}(T), \tilde{\boldsymbol{\theta}}_{i-1}(T)) + \frac{1}{2}\sigma_i^2(0) - \frac{1}{2}\sigma_{i-1}^2(T) \\
&\quad -(\beta - c) \int_0^T \sigma_i^2 dt + \frac{1}{4c} \int_0^T e_i^2 dt.
\end{aligned}
$$

Repeating the operation $i-1$ times leads to the following

$$
\begin{aligned}
V(\sigma_i(T), \tilde{\boldsymbol{\theta}}_i(T)) &\le V(\sigma_1(T), \tilde{\boldsymbol{\theta}}_1(T)) + \frac{1}{2}\sum_{j=2}^{i} \sigma_j^2(0) - \frac{1}{2}\sum_{j=2}^{i} \sigma_{j-1}^2(T) \\
&\quad -(\beta - c) \sum_{j=2}^{i} \int_0^T \sigma_j^2 dt + \frac{1}{4c} \sum_{j=2}^{i} \int_0^T e_j^2 dt
\end{aligned} \tag{5.24}
$$

Part 1) From the initial conditions a), b) and e), we have

$$
\frac{1}{2}\sum_{j=2}^{i} \sigma_j^2(0) - \frac{1}{2}\sum_{j=2}^{i} \sigma_{j-1}^2(T) \le \frac{1}{2}\sigma_0
$$

and (5.24) becomes

$$
\begin{aligned}
V(\sigma_i(T), \tilde{\boldsymbol{\theta}}_i(T)) &\le V(\sigma_1(T), \tilde{\boldsymbol{\theta}}_1(T)) + \frac{1}{2}\sigma_0 - (\beta - c)\sum_{j=2}^{i} \int_0^T \sigma_j^2 dt \\
&\quad + \frac{1}{4c} \sum_{j=2}^{i} \int_0^T e_j^2 dt
\end{aligned} \tag{5.25}
$$

To derive the convergence, the reduction to absurdity will be used. Suppose, on the contrary, there exists a positive integer N_1 such that $\|\sigma_j\|_T \ge \epsilon$ for all iteration number $j \ge N_1$. Since $e_j(\mathbf{x}_j)$ is a convergent sequence in $\mathcal{L}_T^2$, for arbitrary given ϵ, there exists a positive integer N_2 such that $\int_0^T e_j^2 dt \le 2c(\beta - c)T\epsilon$ for all $j \ge N_2$. Let $N = \max\{N_1, N_2\}$, and notice the existence of solution shown in Theorem 5.1, the following quantity is finite

$$
B \triangleq V(\sigma_1(T), \tilde{\boldsymbol{\theta}}_1(T)) + \frac{1}{2}\sigma_0 - (\beta - c)\sum_{j=2}^{N} \int_0^T \sigma_j^2 dt + \frac{1}{4c}\sum_{j=2}^{N} \int_0^T e_j^2 dt.
$$

Then it follows from (5.25) that

$$\begin{aligned} V(\sigma_i(T), \boldsymbol{\theta}_i(T)) &\leq B - (\beta - c) \sum_{j=N+1}^{i} \int_0^T \sigma_j^2 dt + \frac{1}{4c} \sum_{j=N+1}^{i} \int_0^T e_j^2 dt \\ &\leq B - (\beta - c)T(i-N)(\epsilon - \frac{\epsilon}{2}) \\ &= B - \frac{1}{2}(\beta - c)T(i-N)\epsilon. \end{aligned} \tag{5.26}$$

When $i \to \infty$, the right hand side of (5.26) approaches $-\infty$ since B is finite, which contradict the fact that $V(\sigma_i(T), \tilde{\boldsymbol{\theta}}_i(T))$ is positive definite. Therefore, there must exist a subsequence of σ_i which enters the given bound ϵ after a finite number of learning iterations.

Part 2) The relation (5.24) with the initial conditions c) and d), $|\sigma_i(0)| \leq \sigma_0$, is

$$\begin{aligned} V(\sigma_i(T), \tilde{\boldsymbol{\theta}}_i(T)) \leq\ & V(\sigma_1(T), \tilde{\boldsymbol{\theta}}_1(T)) + \frac{1}{2} \sum_{j=2}^{i} \sigma_0^2 \\ & -(\beta - c) \sum_{j=2}^{i} \int_0^T \sigma_j^2 dt + \frac{1}{4c} \sum_{j=2}^{i} \int_0^T e_j^2 dt \end{aligned} \tag{5.27}$$

Analogous to Part 1) proof, assume that there exists a positive integer N_1 such that $\|\sigma_j\|_T \geq \epsilon$ for all iteration number $j \geq N_1$. Since the approximation error e_i is a convergent sequence in $\mathcal{L}_T^2$, there exists an integer N_2 such that $\int_0^T e_j^2 dt \leq 2c(\beta - c)T\epsilon$ for all $j \geq N_2$. From the existence of solution and the finiteness of $N = \max\{N_1, N_2\}$,

$$B \triangleq V(\sigma_1(T), \tilde{\boldsymbol{\theta}}_1(T)) + \frac{1}{2} N \sigma_0^2 - (\beta - c) \sum_{j=2}^{N} \int_0^T \sigma_j^2 dt + \frac{1}{4c} \sum_{j=2}^{N} \int_0^T e_j^2 dt$$

is a finite. For arbitrary $\delta > 0$ and $\epsilon = \frac{\sigma_0^2 + \delta}{(\beta - c)T}$, substitution into (5.27) yields

$$\begin{aligned} V(\sigma_i(T), \tilde{\boldsymbol{\theta}}_i(T)) &\leq B + \frac{1}{2} \sum_{j=N+1}^{i} \sigma_0^2 - (\beta - c) \sum_{j=N+1}^{i} \int_0^T \sigma_j^2 dt + \frac{1}{4c} \sum_{j=N+1}^{i} \int_0^T e_j^2 dt \\ &\leq B + \frac{1}{2} \sum_{j=N+1}^{i} [\sigma_0^2 - (\beta - c)T\epsilon] \\ &= B - \frac{1}{2}(i - N)\delta. \end{aligned} \tag{5.28}$$

The right hand side of (5.28) approaches $-\infty$ because B is finite, which leads to a contradiction to the fact $V(\sigma_i(T), \tilde{\boldsymbol{\theta}}_i(T))$ is positive definite. Therefore, there must exist a subsequence of σ_i which enters the given bound ϵ after a finite number of learning iterations. □

Remark 5.1. *From Part 2 of Theorem 5.2, a large gain β can reduce the tracking error bound ϵ under the initial conditions c) and d).*

Remark 5.2. *It should be noted that in deriving the above convergence properties, we consider only sufficient conditions or the worst case performance. In practice, we may achieve better learning performance such as pointwise or uniform convergence, although in theory only $\mathcal{L}_T^2$ convergence is guaranteed.*

5.4 Robust Adaptive Learning Control

In Section 5.3, we studied the adaptive learning control problem with the unknown function $f(\mathbf{x}) \in \mathcal{L}^2(R^n)$. However, functions in the space $\mathcal{L}^2(R^n)$ are rarely met in practice. For instance, a simple linear function $f(\mathbf{x}) = \mathbf{x}$ does not belong to the space. In this section, our objective is to study functions more general than $\mathcal{L}^2(R^n)$. As such we consider functions in $\mathcal{L}^2(\mathcal{D})$ where $\mathcal{D} \subset R^n$ is a compact set. Most functions we handle in control practice belong to $\mathcal{L}^2(\mathcal{D})$. Comparing with $\mathcal{L}^2(R^n)$, the difficulty of function approximation for $\mathcal{L}^2(\mathcal{D})$ is that the basis defined on $\mathcal{D}$ will not be valid outside the compact set $\mathcal{D}$. In particular the weights $\boldsymbol{\theta}$ will change when the states $\mathbf{x}$ move out the compact set $\mathcal{D}$. Most of function approximation based control methods developed hitherto require the system states to strictly stay in $\mathcal{D}$, or no expansion from $\mathcal{D}$. Such a non-expansion condition in fact is concerned with the transient behavior of control systems and is in general far more difficult than the original control task of asymptotic convergence. On the other hand, robust control methods can easily constrain the system states in $\mathcal{D}$ all the time, provided the unknown functions satisfy Assumption 5.2. Most studies on robust control are based on this assumption. In this section, we study the possibility of combining robust control with the function approximation to achieve better control performance for the plant S_I.

It is well known that in robust control, to achieve a small tracking error bound in the presence of non-vanishing perturbations a high feedback gain is required. The smaller the error bound, the higher the gain. Using an over large control gain will however incur excessive control actions, not only wasting energy but also degrading responses, shortening the life cycle of control mechanisms, or even destabilizing the control system. An appropriate control approach is to incorporate function approximation into robust control. The robust control with a lower gain will guarantee a bounded tracking performance, say $\mathcal{D}$, although the error bound may not meet the performance specification. Then the function approximation with adaptive learning will gradually take over the tracking task by generating necessary control signals to compensate any non-vanishing perturbations or produce the "internal model".

Consider a compact set

$$\mathcal{D}_0 = \{\sigma_i \in R : |\sigma_i|_s \leq \epsilon_0\}, \tag{5.29}$$

where $\epsilon_0 > 0$ is a sufficiently large constant so that the initial conditions $|\sigma(0)| \leq \sigma_0$ is within the compact set. From the definition of the augmented tracking error $\sigma_i(t)$ in (5.6), corresponding to $\mathcal{D}_0$ there exists a compact set $\mathcal{D}$ so that $\mathbf{x}_i \in \mathcal{D}$. As far as we can prove the non-expansion property of the compact set $\mathcal{D}_0$ for any i and $t \in [0,\ T]$, then the non-expansion property of $\mathcal{D}$ is guaranteed. The non-expansion of $\mathcal{D}$ warrants a valid function approximation sequence because the weights $\boldsymbol{\theta}$ will not change. To fulfill this control task, we need to show two properties in the robust adaptive learning control (RALC): the first to show the non-expansion of

$\mathcal{D}_0$, namely the boundeness of σ_i by ϵ_0; and the second to show the convergence of the tracking error sequence $\|\sigma_i\|_T$ to the pre-specified bound ϵ.

In the preceding section we have shown the learning convergence analysis for $f \in \mathcal{L}^2(R^n)$. In order to make use of the analysis results in Theorems 5.1 and 5.2, we can modify the functions $f \in \mathcal{L}^2(\mathcal{D})$ into functions of $f^a \in \mathcal{L}^2(R^n)$ defined below

$$f^a(\mathbf{x}) = \begin{cases} f(\mathbf{x}), & |\mathbf{x}|_s \leq \mathcal{D}, \\ 0, & |\mathbf{x}|_s \geq 2\mathcal{D}, \end{cases}$$

and further let $f^a(\mathbf{x})$ be smooth and monotone between the boundaries $\partial\mathcal{D}$ and $\partial 2\mathcal{D}$. The following figure shows the idea. It is obvious that $f^a(\mathbf{x}) \in \mathcal{L}^2(R^n)$ and

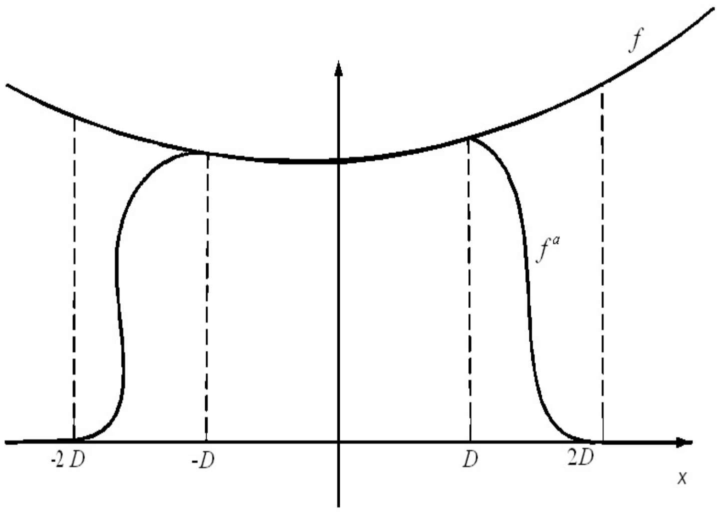

Figure 5.2. The relationship between $f(\mathbf{x})$ and $f^a(\mathbf{x})$

$f(\mathbf{x}) = f^a(\mathbf{x})$ for $\mathbf{x} \in \mathcal{D}$.

Remark 5.3. *Note that such a modification is fictitious, because the states $\mathbf{x}$ will not leave $\mathcal{D}$ by the robust control part, as we will show later. Hence the construction of such a fictitious f^a is only for the convenience of analysis. Likewise, the bounding function $\bar{f}$ of f, defined on $\mathcal{D}$, can also be modified into a fictitious $\bar{f}^a$ defined on R^n, with $\bar{f}^a = \bar{f}$ where $\mathbf{x} \in \mathcal{D}$.*

Now we are ready to construct an augmented plant

$$S_a : \begin{cases} \dot{x}_j = x_{j+1}, & j = 1, 2, \cdots, n-1, \\ \dot{x}_n = f^a(\mathbf{x}) + u & \mathbf{x}(0) = \mathbf{x}_0 \end{cases} \tag{5.30}$$

which has the same form as S_I. The ALC law (5.12) will be revised with an additional robust control, β_i, as follows

$$\begin{aligned} u_i &= -(\beta + \beta_i)\sigma_i - \hat{\boldsymbol{\theta}}_i^T \mathbf{g}_i - v_i, \\ \beta_i &= \frac{|\hat{\boldsymbol{\theta}}_i^T \mathbf{g}_i| + \bar{f}_i^a}{\epsilon_0}, \end{aligned} \tag{5.31}$$

where $\beta > 1$, and $\hat{\boldsymbol{\theta}}_i^T \mathbf{g}_i$ is the function approximation series of f^a on R^n.

Substituting the RALC law (5.31), the dynamics of the tracking error σ_i is

$$\dot{\sigma}_i = -(\beta + \beta_i)\sigma_i - \hat{\boldsymbol{\theta}}_i^T \mathbf{g}_i + f_i^a \tag{5.32}$$

where $f_i^a = f^a(\mathbf{x}_i)$. In the following we derive the non-expansion property of the robust adaptive learning control system.

Theorem 5.3. *For the plant S_a shown in (5.30) satisfying Assumption 5.4, the controller (5.31) together with the parametric adaptive learning law (5.13) guarantees $\sigma_i \in \mathcal{D}_0$ for any i and $t \in [0,\ T]$.*

Proof. Differentiating the following Lyapunov function

$$V(\sigma_i) = \frac{1}{2}\sigma_i^2 \tag{5.33}$$

with respect to time t, substituting the tracking error dynamics (5.32) and the control law (5.31), we have

$$\begin{aligned} \dot{V}(\sigma_i) &= \sigma_i[-(\beta + \beta_i)\sigma_i - \hat{\boldsymbol{\theta}}_i^T \mathbf{g}_i + f_i^a] \\ &\le -\beta\sigma_i^2 - \beta_i|\sigma_i|(|\sigma_i| - \frac{|\hat{\boldsymbol{\theta}}_i^T \mathbf{g}_i| + |\bar{f}_i^a|}{\beta_i}) \\ &= -\beta\sigma_i^2 - \beta_i|\sigma_i|(|\sigma_i| - \epsilon_0). \end{aligned} \tag{5.34}$$

Clearly $\dot{V}$ is negative definite if $|\sigma_i| \ge \epsilon_0$, thus $|\sigma_i(t)| \le \epsilon_0$ is strictly guaranteed for any i and $t \in [0,\ T]$. This implies $\sigma_i \in \mathcal{D}_\sigma$ and $\mathbf{x}_i \in \mathcal{D}$. ☐

Now we are in a position to derive the convergence property of the robust adaptive learning control for the plant S_a.

Theorem 5.4. *For the plant S_a in (5.30), the controller (5.31) together with parametric adaptive learning law (5.13) guarantee the existence of a subsequence, $\|\sigma_{i_j}\|_T$ of $\|\sigma_i\|_T$, which enters the bound ϵ after a finite number of learning iterations.*

Proof. The idea of the proof is similar to Theorems 5.1 and 5.2. Define the same Lyapunov function

$$V(\sigma_i, \tilde{\boldsymbol{\theta}}_i) = \frac{1}{2}\sigma_i^2 + \frac{1}{2}\tilde{\boldsymbol{\theta}}_i^T \tilde{\boldsymbol{\theta}}_i. \tag{5.35}$$

Differentiating $V(\sigma_i, \tilde{\boldsymbol{\theta}}_i)$ with respect to time t, substituting the tracking error dynamics (5.32) and adaptive learning law (5.13) yield

$$\begin{aligned} \dot{V}(\sigma_i, \tilde{\boldsymbol{\theta}}_i) &= \sigma_i \dot{\sigma}_i - \tilde{\boldsymbol{\theta}}_i^T \dot{\hat{\boldsymbol{\theta}}}_i \\ &= \sigma_i[-(\beta + \beta_i)\sigma_i - \hat{\boldsymbol{\theta}}_i^T \mathbf{g}_i + \boldsymbol{\theta}_i^T \mathbf{g}_i + e_i] - \tilde{\boldsymbol{\theta}}_i^T \mathbf{g}_i \sigma_i \\ &\leq -\beta\sigma_i^2 + \sigma_i e_i. \end{aligned} \tag{5.36}$$

Note that the above relation is the same as (5.19). Thus all subsequent derivations in Theorems 5.1 and 5.2 are valid, hence the convergence property concluded in Theorem 5.2 also holds. □

Remark 5.4. *Any smooth functions can be chosen in the region between $\mathcal{D}$ and $2\mathcal{D}$, and the function approximation result is independent of such a choice.*

Remark 5.5. *By choosing a sufficiently large ϵ_0 that is reciprocal to the robust control gain, the robust control efforts can be greatly reduced. At the same time, the control objective can still be achieved after adaptive learning.*

5.5 Two Extensions

Two extensions will be considered: the first is an extension to the plant S_{II} in (5.3) with partially unknown input coefficient, and the second is an extension to the plant S_{III} in (5.5) which is a cascade dynamics with unmatched components.

5.5.1 Plant with Unknown Input Coefficient

Consider the plant S_{II}. The presence of the partially unknown input coefficient $b(\mathbf{x})$ makes the control task much more difficult to address. Note that if $b(\mathbf{x})$ is a known nonsingular function, the control problem is trivial because we can simply multiply the preceding adaptive learning control law by a factor $b^{-1}(\mathbf{x})$.

Let σ_i be defined the same as (5.6). The tracking error dynamics at the i-th iteration can be expressed as

$$\dot{\sigma}_i = f_i + b_i u_i + v_i, \quad \forall t \in [0,\ T]. \tag{5.37}$$

To facilitate later derivations, we introduce two new quantities. Denote $b_i = b(\mathbf{x}_i) = b(\mathbf{x}_i^0, \sigma_i + v_i^0)$, where $\mathbf{x}_i^0 = [x_{1,i}, \cdots, x_{n-1,i}]^T$, $x_{n,i} = \sigma_i + v_i^0$, and $v_i^0 = x_r^{(n-1)}(t) - [\boldsymbol{\lambda}\ 0]\Delta\mathbf{x}_i$. Then a new quantity is defined below

$$w(\boldsymbol{\chi}_i) = \frac{1}{\sigma_i}\int_0^{\sigma_i} [s \sum_{j=1}^{n-1} \frac{\partial b^{-1}(\mathbf{x}_i^0, s + v_i^0)}{\partial x_{j,i}} x_{j+1,i} + b^{-1}(\mathbf{x}_i^0, s + v_i^0) v_i] ds, \tag{5.38}$$

where $\boldsymbol{\chi}_i = [\mathbf{x}_i^T, \sigma_i, v_i, v_i^0]^T \in R^{n+3}$. Another new quantity is

$$\eta_i = \frac{f_i}{b_i}$$

which is nonsingular because $b_i \geq b_0 > 0$ according to Assumption 5.2.

Analogous to Section 5.5, choose a compact set $\mathcal{D}_0 \subset R$ defined by (5.29), assume that a robust controller can make $\sigma_i \in \mathcal{D}_0$ strictly for any i and $t \in [0,\ T]$. Then corresponding to $\mathcal{D}_0$ there exist a compact set $\mathcal{D} \subset R^n$ so that $\mathbf{x}_i \in \mathcal{D}$, and a compact set $\mathcal{D}_1 \subset R^{n+3}$ so that $\boldsymbol{\chi}_i \in \mathcal{D}_1$. The properties $\eta_i \in \mathcal{L}^2(\mathcal{D})$ and $w_i \in \mathcal{L}^2(\mathcal{D}_1)$ are straightforward. Further, following the same idea shown in Figure 5.2, functions η_i and w_i can be modified to be $\mathcal{L}^2(R^n)$ and $\mathcal{L}^2(R^{n+3})$ respectively.

Being in $\mathcal{L}^2$ space, there exist bases $g_1(\mathbf{x}), g_2(\mathbf{x}), \cdots$ and $w_1(\boldsymbol{\chi}), w_2(\boldsymbol{\chi}), \cdots$, all continuous and locally Lipschitz continuous, such that the following functions approximation hold

$$\begin{aligned}\eta^a(\mathbf{x}) &= \sum_{k=1}^{\infty} \theta_k g_k(\mathbf{x}),\\ w^a(\boldsymbol{\chi}) &= \sum_{k=1}^{\infty} \phi_k w_k(\boldsymbol{\chi}),\end{aligned}$$

with unique weights θ_k and ϕ_k. Denote the approximation errors $e_i^{\eta} = \eta(\mathbf{x}_i) - \sum_{k=1}^{i} \theta_k g_k(\mathbf{x}_i)$ and $e_i^w = w(\boldsymbol{\chi}_i) - \sum_{k=1}^{i} \phi_k w_k(\boldsymbol{\chi}_i)$. By choosing bases to be sufficiently smooth and well localized as discussed in Section 5.3, the approximation error sequences, e_i^{η} and e_i^w, will also be convergent in $\mathcal{L}_T^2$ norm as $i \to \infty$.

The robust adaptive learning control mechanism is given below

$$u_i = -(\beta + \beta_i)\sigma_i - \hat{\boldsymbol{\theta}}_i^T \mathbf{g}_i - \hat{\boldsymbol{\phi}}_i^T \mathbf{w}_i, \tag{5.39}$$

where $\hat{\boldsymbol{\theta}}_i = [\hat{\theta}_1, \cdots, \hat{\theta}_i]^T$, $\hat{\boldsymbol{\phi}}_i = [\hat{\phi}_1, \cdots, \hat{\phi}_i]^T$, $\mathbf{g}_i = [g_1, \cdots, g_i]^T$, $\mathbf{w}_i = [w_1, \cdots, w_i]^T$, $\beta > 1$, and the robust control part is

$$\beta_i = \frac{|\hat{\boldsymbol{\theta}}_i^T \mathbf{g}_i| + |\hat{\boldsymbol{\phi}}_i^T \mathbf{w}_i| + \bar{f}_i/b_0 + |v_i|/b_0}{\epsilon_0}.$$

The parametric adaptive learning law is

$$\begin{aligned}\dot{\hat{\boldsymbol{\theta}}}_i &= \sigma_i \mathbf{g}_i, \quad \hat{\boldsymbol{\theta}}_1(0) = 0, \quad \hat{\boldsymbol{\theta}}_i(0) = \hat{\boldsymbol{\theta}}_{i-1}(T),\\ \dot{\hat{\boldsymbol{\phi}}}_i &= \sigma_i \mathbf{w}_i, \quad \hat{\boldsymbol{\phi}}_1(0) = 0, \quad \hat{\boldsymbol{\phi}}_i(0) = \hat{\boldsymbol{\phi}}_{i-1}(T).\end{aligned} \tag{5.40}$$

The non-expansion property of $\mathcal{D}_0$ by the RALC law (5.39) is summarized in the following theorem.

Theorem 5.5. *For the dynamic system S_{II} in (5.4) satisfying Assumptions 5.2 and 5.4, the controller (5.39) guarantees $\sigma_i \in \mathcal{D}_0$ for any i and $t \in [0,\ T]$.*

Proof. Substituting the control law (5.39) into the tracking error dynamics (5.37) yields

$$\dot{\sigma}_i = f_i - b_i(\beta + \beta_i)\sigma_i - b_i\hat{\boldsymbol{\theta}}_i^T \mathbf{g}_i - b_i\hat{\boldsymbol{\phi}}_i^T \mathbf{w}_i + v_i. \tag{5.41}$$

Differentiating the following Lyapunov function

$$V(\sigma_i) = \frac{1}{2}\sigma_i^2 \tag{5.42}$$

with respect to time t, substituting the dynamics (5.41), and using the fact $b_i \geq b_0 > 0$, we obtain

$$\begin{aligned} \dot{V}(\sigma_i) &= \sigma_i[f_i - b_i(\beta + \beta_i)\sigma_i - b_i\hat{\boldsymbol{\theta}}_i^T \mathbf{g}_i - b_i\hat{\boldsymbol{\phi}}_i^T \mathbf{w}_i + v_i] \\ &\leq -b_0\beta\sigma_i^2 - b_i\beta_i\sigma_i^2 + b_i|\hat{\boldsymbol{\theta}}_i^T \mathbf{g}_i\sigma_i| + b_i|\hat{\boldsymbol{\phi}}_i^T \mathbf{w}_i\sigma_i| + |\bar{f}_i\sigma_i| + |v_i\sigma_i| \\ &\leq -b_0\beta\sigma_i^2 - b_i\beta_i|\sigma_i|(|\sigma_i| - \frac{|\hat{\boldsymbol{\theta}}_i^T \mathbf{g}_i|\sigma_i| + |\hat{\boldsymbol{\phi}}_i^T \mathbf{w}_i\sigma_i| + |\bar{f}_i\sigma_i|/b_i + |v_i\sigma_i|/b_i}{\beta_i}) \\ &\leq -b_0\beta\sigma_i^2 - b_i\beta_i|\sigma_i|[|\sigma_i| - \epsilon_0]. \end{aligned} \tag{5.43}$$

Clearly $\dot{V}$ is negative definiteness for $|\sigma_i| > \epsilon_0$, hence $\sigma_i \in \mathcal{D}_0$ for any i and $t \in [0,\ T]$. □

The convergence property is summarized below.

Theorem 5.6. *For the plant S_{II} in (5.4), the controller (5.39) together with adaptive learning law (5.40) guarantee that the existence of a subsequence, $\|\sigma_{i_j}\|_T$ of $\|\sigma_i\|_T$, which enters the bound ϵ after a finite number of learning iterations.*

Proof. First, define a smooth scalar function (Zhang *et al.*, 2000)

$$F(\sigma_i) = \int_0^{\sigma_i} sb^{-1}(\mathbf{x}_i^0, s + v_i^0)ds \tag{5.44}$$

which is a function of σ_i, $\mathbf{x}_i^0$ and v_i^0. Based on the mean value theory (Apostol, 1957), $F(\sigma_i)$ can be rewritten as $F(\sigma_i) = c\sigma_i^2 b^{-1}(\mathbf{x}_i^0, c\sigma_i + v_i^0)$ with $c \in (0, 1)$. Since $b^{-1}(\mathbf{x}_i) > 0, \forall\ \mathbf{x}_i \in \mathcal{D}$, it is shown that $F(\sigma_i)$ is positive definitive with respect to σ_i.

Furthermore,

$$\begin{aligned}\dot{F} &= \frac{\partial F}{\partial \sigma_i}\dot{\sigma}_i + \frac{\partial F}{\partial \mathbf{x}_i^0}\dot{\mathbf{x}}_i^0 + \frac{\partial F}{\partial v_i^0}\dot{v}_i^0 \\ &= b^{-1}(\mathbf{x}_i)\sigma_i\dot{\sigma}_i + \int_0^{\sigma_i} s[\frac{\partial b^{-1}(\mathbf{x}_i^0, s+v_i^0)}{\partial \mathbf{x}_i^0}\dot{\mathbf{x}}_i^0]ds + \dot{v}_i^0\int_0^{\sigma_i} s[\frac{\partial b^{-1}(\mathbf{x}_i^0, s+v_i^0)}{\partial v_i}]ds.\end{aligned} \tag{5.45}$$

From the definition of $\mathbf{x}_i^0$, we have

$$\int_0^{\sigma_i} s[\frac{\partial b^{-1}(\mathbf{x}_i^0, s+v_i^0)}{\partial \mathbf{x}_i^0}\dot{\mathbf{x}}_i^0]ds = \int_0^{\sigma_i} s\sum_{j=1}^{n-1}\frac{\partial b^{-1}(\mathbf{x}_i^0, s+v_i^0)}{\partial x_{j,i}}x_{j+1,i}ds. \tag{5.46}$$

Since $\frac{\partial b^{-1}(\mathbf{x}_i^0, s+v_i^0)}{\partial v_i^0} = \frac{\partial b^{-1}(\mathbf{x}_i^0, s+v_i^0)}{\partial s}$, $v_i = -\dot{v}_i^0$, it follows that

$$\begin{aligned}\dot{v}_i^0\int_0^{\sigma_i} s[\frac{\partial b^{-1}(\mathbf{x}_i^0, s+v_i^0)}{\partial v_i^0}]ds &= -v_i\int_0^{\sigma_i} s[\frac{\partial b^{-1}(\mathbf{x}_i^0, s+v_i^0)}{\partial s}]ds \\ &= -v_i[sb^{-1}(\mathbf{x}_i^0, s+v_i^0)|_0^{\sigma_i} - \int_0^{\sigma_i} b^{-1}(\mathbf{x}_i^0, s+v_i^0)ds] \\ &= -b^{-1}(\mathbf{x}_i)v_i\sigma_i + \int_0^{\sigma_i} b^{-1}(\mathbf{x}_i^0, s+v_i^0)v_i ds.\end{aligned} \tag{5.47}$$

Substituting (5.41), (5.46) and (5.47) into (5.45), we obtain

$$\begin{aligned}\dot{F} &= -\beta\sigma_i^2 + \sigma_i(\eta_i - \hat{\boldsymbol{\theta}}_i^T\mathbf{g}_i) \\ &\quad +\sigma_i\left(\frac{1}{\sigma_i}\int_0^{\sigma_i}[s\sum_{j=1}^{n-1}\frac{\partial b^{-1}(\mathbf{x}_i^0, s+v_i^0)}{\partial x_{j,i}}x_{j+1,i} + b^{-1}(\mathbf{x}_i^0, s+v_i^0)v_i]ds - \sigma_i\hat{\boldsymbol{\phi}}_i^T\mathbf{w}_i\right) \\ &= -\beta\sigma_i^2 + \sigma_i\tilde{\boldsymbol{\theta}}_i^T\mathbf{g}_i + \sigma_i e_i^\eta + \sigma_i\tilde{\boldsymbol{\phi}}_i^T\mathbf{w}_i + \sigma_i e_i^w,\end{aligned} \tag{5.48}$$

where $\tilde{\boldsymbol{\theta}} = \boldsymbol{\theta} - \hat{\boldsymbol{\theta}}$ and $\tilde{\boldsymbol{\phi}} = \boldsymbol{\phi} - \hat{\boldsymbol{\phi}}$.

Now choose a Lyapunov function

$$V(\sigma_i, \tilde{\boldsymbol{\theta}}_i, \tilde{\boldsymbol{\phi}}_i) = F + \frac{1}{2}\tilde{\boldsymbol{\theta}}_i^T\tilde{\boldsymbol{\theta}}_i + \frac{1}{2}\tilde{\boldsymbol{\phi}}_i^T\tilde{\boldsymbol{\phi}}_i. \tag{5.49}$$

The time derivative of $V(\sigma_i, \tilde{\boldsymbol{\theta}}, \tilde{\boldsymbol{\phi}})$ is

$$\begin{aligned}\dot{V} &= \dot{F} - \tilde{\boldsymbol{\theta}}_i^T\dot{\hat{\boldsymbol{\theta}}}_i - \tilde{\boldsymbol{\phi}}_i^T\dot{\hat{\boldsymbol{\phi}}}_i \\ &= \dot{F} - \sigma_i\tilde{\boldsymbol{\theta}}_i^T\mathbf{g}_i - \sigma_i\tilde{\boldsymbol{\phi}}_i^T\mathbf{w}_i.\end{aligned} \tag{5.50}$$

Substituting (5.48) into (5.50), it follows that

$$\dot{V} = -\beta\sigma_i^2 + \sigma_i(e_i^\eta + e_i^w)$$

which is almost the same as (5.19) except that approximation term e_i is replaced by an augmented approximation term $e_i^\eta + e_i^w$ that is $\mathcal{L}_T^2$ convergent. Thus all subsequent derivations in Theorems 5.1 and 5.2 are valid, and the convergence property concluded in Theorem 5.2 also holds. □

5.5.2 Plant in Cascade Form

Consider the n-th order cascade dynamic system S_{III} in (5.5). The backstepping design has been developed as a systematic approach to handle cascade dynamics or any systems in triangular form. The principal idea of backstepping design is for the j-th subsystem to construct a fictitious control input, which will enter the $(j+1)$-th subsystem as the objective trajectory. In what follows we will demonstrate the adaptive learning control based on the backstepping design. As a systematic method, the backstepping design can be easily extended from second order to n-th order, hence for simplicity and concentration on the most fundamental steps in the problem solving, we consider a second order dynamics, i.e. $n = 2$ in (5.5) as below

$$\begin{aligned} \dot{x}_{1,i} &= f_1(x_{1,i}) + x_{2,i} \\ \dot{x}_{2,i} &= f_2(\mathbf{x}_i) + u_i \end{aligned} \tag{5.51}$$

where $\mathbf{x}_i = [x_{1,i},\ x_{2,i}]^T$. Denote $f_{1,i} = f_1(x_{1,i})$ and $f_{2,i} = f_1(\mathbf{x}_i)$. The control objective is to design an appropriate control input $u_i(t)$ such that $x_{1,i}$ can track $x_{r,1}$ in $\mathcal{L}_T^2$ as $i \to \infty$.

Since $f_{j,i} \in \mathcal{L}^2(R^j)$ for $j = 1, 2$, there exist continuous and locally Lipschitzian bases $g_i = g(x_{1,i})$ and $h_i = h(\mathbf{x}_i)$ such that

$$\begin{aligned} f_{1,i} &= \sum_{k=1}^{\infty} \theta_k g_k = \sum_{k=1}^{i} \theta_k g_k + e_{1,i} = \boldsymbol{\theta}_i^T \mathbf{g}_i + e_{1,i}, \\ f_{2,i} &= \sum_{k=1}^{\infty} \phi_k h_k = \sum_{k=1}^{i} \phi_k h_k + e_{2,i} = \boldsymbol{\phi}_i^T \mathbf{h}_i + e_{2,i} \end{aligned}$$

where $e_{1,i}$ and $e_{2,i}$ are approximation errors. Define new coordinates $z_{1,i} = x_{1,i} - x_{r,1}$ and $z_{2,i} = x_{2,i} - \alpha_{1,i}$, the fictitious control is

$$\alpha_{1,i} = -\beta_1 z_{1,i} + \dot{x}_{r,1} - \hat{\boldsymbol{\theta}}_i^T \mathbf{g}_i \tag{5.52}$$

where $\beta_1 > 1$, and the parametric adaptive learning law is

$$\begin{aligned} \dot{\hat{\boldsymbol{\theta}}}_i &= \mathbf{g}_i z_{1,i} + \rho_{1,i} \mathbf{g}_i z_{2,i}, \\ \hat{\boldsymbol{\theta}}_1(0) &= 0, \quad \hat{\boldsymbol{\theta}}_i(0) = \hat{\boldsymbol{\theta}}_{i-1}(T), \end{aligned} \tag{5.53}$$

where

$$\rho_{1,i} = \frac{\partial \alpha_{1,i}}{\partial x_{1,i}} + \left(\frac{\partial \alpha_{1,i}}{\partial \mathbf{g}_i} \right)^T \frac{\partial \mathbf{g}_i}{\partial x_{1,i}}.$$

Design the actual controller at $i-$th iteration

$$u_i = \rho_{2,i} - z_{1,i} - \beta_2 z_{2,i} + \rho_{1,i} \hat{\boldsymbol{\theta}}_i^T \mathbf{g}_i - \hat{\boldsymbol{\phi}}_i^T \mathbf{h}_i \tag{5.54}$$

where $\beta_2 > \rho_{1,i}^2 + 1$ and

$$\begin{aligned} \rho_{2,i} = &\frac{\partial \alpha_{1,i}}{\partial t} + \frac{\partial \alpha_{1,i}}{\partial x_{1,i}} x_{2,i} + \frac{\partial \alpha_{1,i}}{\partial x_{r,1}} \dot{x}_{r,1} + \frac{\partial \alpha_{1,i}}{\partial \dot{x}_{r,1}} x_{r,1}^{(2)} \\ &+ \left(\frac{\partial \alpha_{1,i}}{\partial \hat{\boldsymbol{\theta}}_i} \right)^T \dot{\hat{\boldsymbol{\theta}}}_i + \left(\frac{\partial \alpha_{1,i}}{\partial \mathbf{g}_i} \right)^T \frac{\partial \mathbf{g}_i}{\partial x_{1,i}} x_{2,i}. \end{aligned}$$

The second parametric adaptive learning law is

$$\begin{aligned} \dot{\hat{\boldsymbol{\phi}}}_i &= \mathbf{h}_i z_{2,i}, \\ \hat{\boldsymbol{\phi}}_1(0) &= 0, \quad \hat{\boldsymbol{\phi}}_i(0) = \hat{\boldsymbol{\phi}}_{i-1}(T). \end{aligned} \tag{5.55}$$

The convergence property of the above adaptive learning control scheme is derived by the following theorem.

Theorem 5.7. *For the plant (5.51), the control laws (5.52), (5.54) and the adaptive learning laws (5.53) and (5.55) guarantee the existence of a subsequence* $\{z_{1,i_j}\}$ *of* $\{z_{1,i}\}$ *such that for arbitrary* $\epsilon > 0$, $\|z_{1,i_j}\|_T$ *enters the bound* ϵ *after a finite number of learning iterations.*

Proof. The proof consists of two steps.

Step 1.

From (5.51), we have

$$\begin{aligned}\dot{z}_{1,i} &= \dot{x}_{1,i} - \dot{x}_{r,1} \\ &= x_{2,i} + f_{1,i} - \dot{x}_{r,1} \\ &= z_{2,i} + \alpha_{1,i} + f_{1,i} - \dot{x}_{r,1}. \end{aligned} \tag{5.56}$$

Substituting the fictitious control $\alpha_{1,i}$ in (5.52) into (5.56) yields

$$\begin{aligned}\dot{z}_{1,i} &= z_{2,i} - \beta_1 z_{1,i} + f_{1,i} - \hat{\boldsymbol{\theta}}_i^T \mathbf{g}_i \\ &= z_{2,i} - \beta_1 z_{1,i} + \tilde{\boldsymbol{\theta}}_i^T \mathbf{g}_i + e_{1,i}. \end{aligned} \tag{5.57}$$

Define a Lyapunov function below

$$V_{1,i} = \frac{1}{2} z_{1,i}^2 + \frac{1}{2} \tilde{\boldsymbol{\theta}}_i^T \tilde{\boldsymbol{\theta}}_i. \tag{5.58}$$

Using (5.57), the derivative of $V_{1,i}$ is

$$\begin{aligned}\dot{V}_{1,i} &= z_{1,i}\dot{z}_{1,i} - \tilde{\boldsymbol{\theta}}_i^T \dot{\hat{\boldsymbol{\theta}}}_i \\ &= z_{1,i}(z_{2,i} - \beta_1 z_{1,i} + \tilde{\boldsymbol{\theta}}_i^T \mathbf{g}_i + e_{1,i}) - \tilde{\boldsymbol{\theta}}_i^T \dot{\hat{\boldsymbol{\theta}}}_i \\ &= z_{1,i} z_{2,i} - \beta_1 z_{1,i}^2 + \tilde{\boldsymbol{\theta}}_i^T \mathbf{g}_i z_{1,i} + e_{1,i} z_{1,i} - \tilde{\boldsymbol{\theta}}_i^T \dot{\hat{\boldsymbol{\theta}}}_i \\ &= z_{1,i} z_{2,i} - \beta_1 z_{1,i}^2 + e_{1,i} z_{1,i} - \tilde{\boldsymbol{\theta}}_i^T (\dot{\hat{\boldsymbol{\theta}}}_i - \mathbf{g}_i z_{1,i}). \end{aligned} \tag{5.59}$$

Step 2.

From (5.51) and (5.52), we have

$$\begin{aligned}
\dot{z}_{2,i} &= \dot{x}_{2,i} - \dot{\alpha}_{1,i} \\
&= u_i + f_{2,i} - \{\frac{\partial \alpha_{1,i}}{\partial t} + \frac{\partial \alpha_{1,i}}{\partial x_{1,i}}\dot{x}_{1,i} + \frac{\partial \alpha_{1,i}}{\partial x_{r,1}}\dot{x}_{r,1} + \frac{\partial \alpha_{1,i}}{\partial \dot{x}_{r,1}}x_{r,1}^{(2)} \\
&\quad +(\frac{\partial \alpha_{1,i}}{\partial \hat{\boldsymbol{\theta}}_i})^T\dot{\hat{\boldsymbol{\theta}}}_i + (\frac{\partial \alpha_{1,i}}{\partial \mathbf{g}_i})^T\frac{\partial \mathbf{g}_i}{\partial x_{1,i}}\dot{x}_{1,i}\} \\
&= u_i + f_{2,i} - \{\frac{\partial \alpha_{1,i}}{\partial t} + \frac{\partial \alpha_{1,i}}{\partial x_{r,1}}\dot{x}_{r,1} + \frac{\partial \alpha_{1,i}}{\partial \dot{x}_{r,1}}x_{r,1}^{(2)} \\
&\quad +(\frac{\partial \alpha_{1,i}}{\partial \hat{\boldsymbol{\theta}}_i})^T\dot{\hat{\boldsymbol{\theta}}}_i + (\frac{\partial \alpha_{1,i}}{\partial \mathbf{g}_i})^T\frac{\partial \mathbf{g}_i}{\partial x_{1,i}}(x_{2,i} + f_{1,i})\} - \frac{\partial \alpha_{1,i}}{\partial x_{1,i}}(x_{2,i} + f_{1,i}) \\
&= u_i + f_{2,i} - [\frac{\partial \alpha_{1,i}}{\partial x_{1,i}} + (\frac{\partial \alpha_{1,i}}{\partial \mathbf{g}_i})^T\frac{\partial \mathbf{g}_i}{\partial x_{1,i}}]f_{1,i} \\
&\quad -\{\frac{\partial \alpha_{1,i}}{\partial t} + \frac{\partial \alpha_{1,i}}{\partial x_{1,i}}x_{2,i} + \frac{\partial \alpha_{1,i}}{\partial x_{r,1}}\dot{x}_{r,1} + \frac{\partial \alpha_{1,i}}{\partial \dot{x}_{r,1}}x_{r,1}^{(2)} \\
&\quad +(\frac{\partial \alpha_{1,i}}{\partial \hat{\boldsymbol{\theta}}_i})^T\dot{\hat{\boldsymbol{\theta}}}_i + (\frac{\partial \alpha_{1,i}}{\partial \mathbf{g}_i})^T\frac{\partial \mathbf{g}_i}{\partial x_{1,i}}x_{2,i}\} \\
&= u_i + f_{2,i} - \rho_{1,i}f_{1,i} - \rho_{2,i} \qquad (5.60)
\end{aligned}$$

where

$$\rho_{2,i} = \frac{\partial \alpha_{1,i}}{\partial t} + \frac{\partial \alpha_{1,i}}{\partial x_{1,i}}x_{2,i} + \frac{\partial \alpha_{1,i}}{\partial x_{r,1}}\dot{x}_{r,1} + \frac{\partial \alpha_{1,i}}{\partial \dot{x}_{r,1}}x_{r,1}^{(2)} + \left(\frac{\partial \alpha_{1,i}}{\partial \hat{\boldsymbol{\theta}}_i}\right)^T\dot{\hat{\boldsymbol{\theta}}}_i + \left(\frac{\partial \alpha_{1,i}}{\partial \mathbf{g}_i}\right)^T\frac{\partial \mathbf{g}_i}{\partial x_{1,i}}x_{2,i}$$

is known.

Substituting the control law (5.54) into (5.60) yields

$$\begin{aligned}
\dot{z}_{2,i} &= -z_{1,i} - \beta_2 z_{2,i} - \rho_{1,i}(f_{1,i} - \hat{\boldsymbol{\theta}}_i^T\mathbf{g}_i) + (f_{2,i} - \hat{\boldsymbol{\phi}}_i^T\mathbf{h}_i) \\
&= -z_{1,i} - \beta_2 z_{2,i} - \rho_{1,i}\tilde{\boldsymbol{\theta}}_i^T\mathbf{g}_i - \rho_{1,i}e_{1,i} + \tilde{\boldsymbol{\phi}}_i^T\mathbf{h}_i + e_{2,i}. \qquad (5.61)
\end{aligned}$$

Define the Lyapunov function below

$$V_{2,i} = V_{1,i} + \frac{1}{2}z_{2,i}^2 + \frac{1}{2}\tilde{\boldsymbol{\phi}}_i^T\tilde{\boldsymbol{\phi}}_i. \qquad (5.62)$$

The derivative of $V_{2,i}$ is

$$\dot{V}_{2,i} = \dot{V}_{1,i} + z_{2,i}\dot{z}_{2,i} - \tilde{\boldsymbol{\phi}}_i^T\dot{\hat{\boldsymbol{\phi}}}_i. \qquad (5.63)$$

Using (5.61), we have

$$\begin{aligned} z_{2,i}\dot{z}_{2,i} &= z_{2,i}\left(-z_{1,i} - \beta_2 z_{2,i} - \rho_{1,i}\tilde{\boldsymbol{\theta}}_i^T \mathbf{g}_i - \rho_{1,i}e_{1,i} + \tilde{\boldsymbol{\phi}}_i^T \mathbf{h}_i + e_{2,i}\right) \\ &= -z_{1,i}z_{2,i} - \beta_2 z_{2,i}^2 - \rho_{1,i}\tilde{\boldsymbol{\theta}}_i^T \mathbf{g}_i z_{2,i} - \rho_{1,i}e_{1,i}z_{2,i} + \tilde{\boldsymbol{\phi}}_i^T \mathbf{h}_i z_{2,i} + e_{2,i}z_{2,i} \end{aligned} \tag{5.64}$$

Substituting (5.59) and (5.64) into (5.63) yields

$$\begin{aligned} \dot{V}_{2,i} &= [z_{1,i}z_{2,i} - \beta_1 z_{1,i}^2 + e_{1,i}z_{1,i} - \tilde{\boldsymbol{\theta}}_i^T(\dot{\hat{\boldsymbol{\theta}}}_i - \mathbf{g}_i z_{1,i})] \\ &\quad +[-z_{1,i}z_{2,i} - \beta_2 z_{2,i}^2 + \rho_{1,i}\tilde{\boldsymbol{\theta}}_i^T \mathbf{g}_i z_{2,i} + \rho_{1,i}e_{1,i}z_{2,i} + \tilde{\boldsymbol{\phi}}_i^T \mathbf{h}_i z_{2,i} + e_{2,i}z_{2,i}] - \tilde{\boldsymbol{\phi}}_i^T \dot{\hat{\boldsymbol{\phi}}}_i \\ &= -\beta_1 z_{1,i}^2 - \beta_2 z_{2,i}^2 - \tilde{\boldsymbol{\theta}}_i^T(\dot{\hat{\boldsymbol{\theta}}}_i - \mathbf{g}_i z_{1,i} - \rho_{1,i}\mathbf{g}_i z_{2,i}) - \tilde{\boldsymbol{\phi}}_i^T(\dot{\hat{\boldsymbol{\phi}}}_i - \mathbf{h}_i z_{2,i}) \\ &\quad +e_{1,i}z_{1,i} + \rho_{1,i}e_{1,i}z_{2,i} + e_{2,i}z_{2,i}. \end{aligned} \tag{5.65}$$

Substitution of the adaptive learning laws (5.53) and (5.55) results in

$$\dot{V}_{2,i} \leq -\beta_1 z_{1,i}^2 - \beta_2 z_{2,i}^2 + e_{1,i}z_{1,i} + \rho_{1,i}e_{1,i}z_{2,i} + e_{2,i}z_{2,i}. \tag{5.66}$$

Using Young's inequality, there exists $c \in (0, 1)$ such that

$$\begin{aligned} e_{1,i}z_{1,i} &\leq cz_{1,i}^2 + \frac{1}{4c}e_{1,i}^2, \\ \rho_{1,i}e_{1,i}z_{2,i} &\leq c\rho_{1,i}^2 z_{2,i}^2 + \frac{1}{4c}e_{1,i}^2 \\ e_{2,i}z_{2,i} &\leq cz_{2,i}^2 + \frac{1}{4c}e_{2,i}^2. \end{aligned}$$

Choosing $(\beta_1 - c) \geq \beta$ and $(\beta_2 - c\rho_{1,i}^2 - c) \geq \beta$ with $\beta > 0$, we obtain

$$\begin{aligned} \dot{V}_{2,i} &\leq -(\beta_1 - c)z_{1,i}^2 - (\beta_2 - c\rho_{1,i}^2 - c)z_{2,i}^2 + \frac{1}{2c}e_{1,i}^2 + \frac{1}{4c}e_{2,i}^2 \\ &\leq -\beta(\sqrt{z_{1,i}^2 + z_{2,i}^2})^2 + (\sqrt{(e_{1,i}^2 + e_{2,i}^2)/2c})^2 \end{aligned}$$

By viewing $\sqrt{z_{1,i}^2 + z_{2,i}^2}$ and $\sqrt{(e_{1,i}^2 + e_{2,i}^2)/2c}$ as lumped quantities, the above relation is analogous to the relation (5.21) in Theorem 5.1. Further $\sqrt{(e_{1,i}^2 + e_{2,i}^2)/2c}$ is convergent in $\mathcal{L}_T^2$ when $i \to \infty$. Therefore by following derivation procedures in Theorems 5.1 and 5.2, we can reach the conclusion that $\|z_{1,i_j}\|_T \leq \epsilon$ can be achieved after a finite number of learning iterations. □

5.6 Wavelet Bases

From previous discussions, finding an appropriate basis is indispensable in order to achieve the desirable function approximation property in ALC or RALC. In this section, we will illustrate how an orthonormal basis of wavelets for $\mathcal{L}^2(R)$ can be constructed from the multiresolution approximation.

5.6.1 Multiresolution Approximations by Wavelet

Multi-resolution analysis was proposed in (Mallat, 1989). Multi-resolution analysis provides a mathematical tool to describe the increment in information from a coarse resolution approximation to a finer resolution approximation. Let us give the definition of this concept. Denote $\mathcal{Z}$ the set of integer numbers.

Definition 5.2. *A multiresolution analysis of $\mathcal{L}^2(R)$ is an increasing sequence $\mathcal{V}_j \in \mathcal{L}^2(R)$, $j \in \mathcal{Z}$, of closed subspaces of $\mathcal{L}^2(R)$, with the following properties*

1. $\cdots \mathcal{V}_{-2} \subset \mathcal{V}_{-1} \subset \mathcal{V}_0 \subset \mathcal{V}_1 \subset \mathcal{V}_2 \cdots$

2. $\bigcap_{-\infty}^{\infty} \mathcal{V}_j = \{\mathbf{0}\}, \quad \bigcup_{-\infty}^{\infty} \mathcal{V}_j = \mathcal{L}^2(R)$ *is dense in* $\mathcal{L}^2(R)$

3. $\forall f \in \mathcal{L}^2(R), \ \forall j \in \mathcal{Z}, \qquad f(x) \in \mathcal{V}_j \Leftrightarrow f(2x) \in \mathcal{V}_{j+1}$

4. $f(x) \in \mathcal{V}_j \Rightarrow f(x - 2^{-j}k) \in \mathcal{V}_j \quad j, k \in \mathcal{Z}$

5. *For all j, there exists a $\phi(x)$, called scaling function, such that $\{\phi_{j,k}(x) = 2^{j/2}\phi(2^j x - k) |\ k \in \mathcal{Z}\}$ is an orthonormal basis of $\mathcal{V}_j$ and $\mathcal{V}_j = \mathrm{span}\,\{\phi_{j,k} |\ k \in \mathcal{Z}\}$.*

The orthogonal projection of a function $f \in \mathcal{L}^2(R)$ into $\mathcal{V}_j$ is given by

$$f_j(x) = \sum_{k \in \mathcal{Z}} < \phi_{j,k}(x), f(x) > \phi_{j,k}(x) \tag{5.67}$$

and can be interpreted as an approximation to f at resolution 2^{-j}. Therefore, the function $f(x)$ can be uniquely approximated in the space $\mathcal{V}_j$

$$\begin{aligned} f(x) &= f_j(x) + e_j \\ &= \sum_{k=1}^{N_j} < \phi_{j,k}(x), f(x) > \phi_{j,k}(x) + e_j \end{aligned}$$

where e_j is the approximation error at j-th resolution including the truncation error, N_j is the number of bases used at the j-th resolution, and $< \cdot >$ is the inner product. Note that a larger j means a higher resolution, therefore $\|e(j+1)\| \leq \|e(j)\|$ and $\lim_{j\to\infty} \|e(j)\| = 0$.

By defining $\mathcal{W}_j$ as the orthogonal complement of $\mathcal{V}_j$ in $\mathcal{V}_{j+1}$, i.e.,

$$\mathcal{V}_{j+1} = \mathcal{V}_j \bigoplus \mathcal{W}_j, \tag{5.68}$$

the space $\mathcal{L}^2(R)$ is represented as a direct sum

$$|\mathcal{L}^2(R) = \bigoplus_{j\in\mathcal{Z}} \mathcal{W}_j. \tag{5.69}$$

Moreover, from the previous assumption on $\mathcal{V}_j$ it follows that there exists a function $\psi(x)$, called mother wavelet, such that

$$\{\psi_{j,k}(x) = 2^{j/2}\psi(2^j x - k) |\ k \in \mathcal{Z}\} \tag{5.70}$$

is an orthonormal basis of $\mathcal{W}_j$. From (5.69), $\{\psi_{j,k}|\ j,k \in \mathcal{Z}\}$ constitutes an orthonormal basis for $\mathcal{L}^2(R)$. The spaces $\mathcal{W}_j$ are called wavelet subspaces of $\mathcal{L}^2(R)$ relative to the scaling function $\phi(x)$ and the orthogonal projection of a function $f \in \mathcal{L}^2(R)$ into $\mathcal{W}_j$, given by

$$g_j(x) = \sum_{k\in\mathcal{Z}} < \psi_{j,k}(x), f(x) > \psi_{j,k}(x) \tag{5.71}$$

can be interpreted as an approximation to f at resolution 2^{-j}. Therefore, the function $f(x)$ in the space $\mathcal{L}^2(R)$ can be uniquely approximated

$$\begin{aligned} f(x) &= \sum_{k\in\mathcal{Z}} < \phi_{J,k}(x), f(x) > \phi_{J,k}(x) + \sum_{j\geq J}\sum_{k\in\mathcal{Z}} < \psi_{j,k}(x), f(x) > \psi_{j,k}(x) \\ &= \sum_{k\in\mathcal{Z}} v_{J,k}\phi_{J,k}(x) + \sum_{j\geq J}\sum_{k\in\mathcal{Z}} w_{j,k}\psi_{j,k}(x) \end{aligned}$$

where $v_{J,k}$ and $w_{j,k}$ denote the coefficients or weights of the wavelet network. For notational convenience, we will drop the subscripts J from the lowest resolution, i.e., $v_{J,k} \to v_k$, and $\phi_{J,k} \to \phi_k$.

5.6.2 Three Wavelet Bases

Let us introduce three different kinds of wavelet bases.

Case 1. Orthonormal Wavelet db3

In Daubechies (Daubechies, 1988) a number of orthonormal bases of wavelets were constructed with compact support. Among them the orthonormal wavelet base, db3, is popular because of its balance between the simplicity of algorithm and smoothness of function approximation. db3 has been widely used in the field signal processing. The scaling function of db3 wavelet is shown below with 6 coefficients

$$\begin{aligned} \phi(x) &= \sqrt{2}[h_0\phi(2x) + h_1\phi(2x-1) + h_2\phi(2x-2) + h_3\phi(2x-3) \\ &\quad + h_4\phi(2x-4) + h_5\phi(2x-5)], \end{aligned}$$

and the coefficients $h_0, \cdots, h_5$ can be solved via the following set of equations

$$\begin{aligned} h_0^2 + h_1^2 + h_2^2 + h_3^2 + h_4^2 + h_5^2 &= 1, \\ h_0h_2 + h_1h_3 + h_2h_4 + h_3h_5 &= 0, \\ h_0h_4 + h_1h_5 &= 0, \\ h_0 - h_1 + h_2 - h_3 + h_4 - h_5 &= 0, \\ -h_1 + 2h_2 - 3h_3 + 4h_4 - 5h_5 &= 0, \\ -h_1 + 4h_2 - 9h_3 + 16h_4 - 25h_5 &= 0. \end{aligned} \tag{5.72}$$

The corresponding wavelet function $\psi(x)$ is defined as

$$\begin{aligned} \psi(x) &= \sqrt{2}[-h_0\phi(2x-1) + h_1\phi(2x) - h_2\phi(2x+1) + h_3\phi(2x+2) \\ &\quad - h_4\phi(2x+3) + h_5\phi(2x+4)]. \end{aligned}$$

db3 scaling function ϕ and wavelet function ψ are shown in Figure 5.3 and Figure 5.4 respectively. Clearly db3 is not smooth, hence might not be an ideal choice

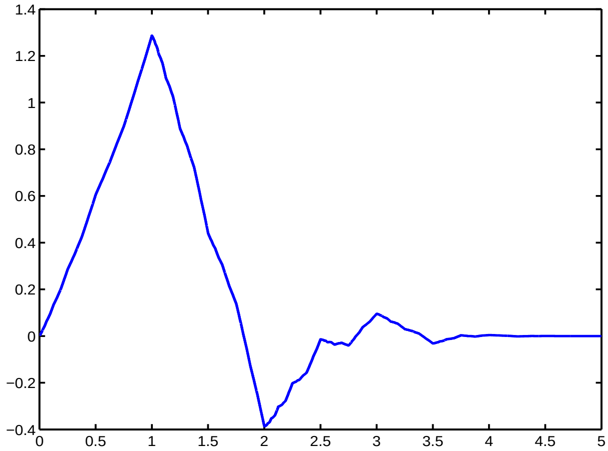

Figure 5.3. Scaling function ϕ of db3

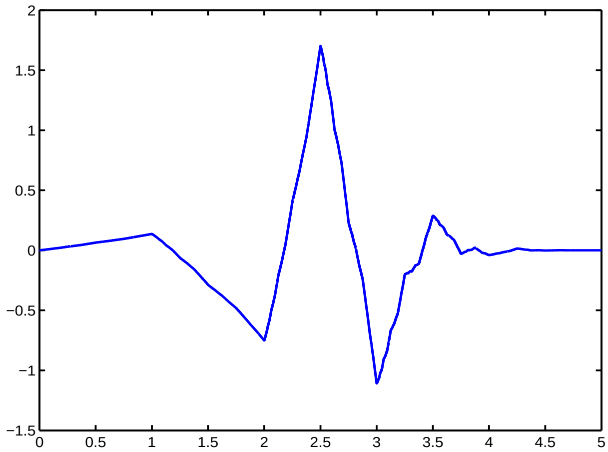

Figure 5.4. Wavelet function ψ of db3

for control problems.

Case 2. Sinc-wavelet

Sinc-wavelet is also widely used to solve signal processing problems. The scaling function of the sinc-wavelet is $\phi(x) = sinc(\pi x)$. The corresponding wavelet function is $\psi(x) = \frac{cos\pi x - sin2\pi x}{\pi(\frac{1}{2}-x)}$. The scaling function ϕ and the wavelet function ψ are shown in Figure 5.5 and Figure 5.6 respectively. Sinc-wavelet is smooth, hence can be

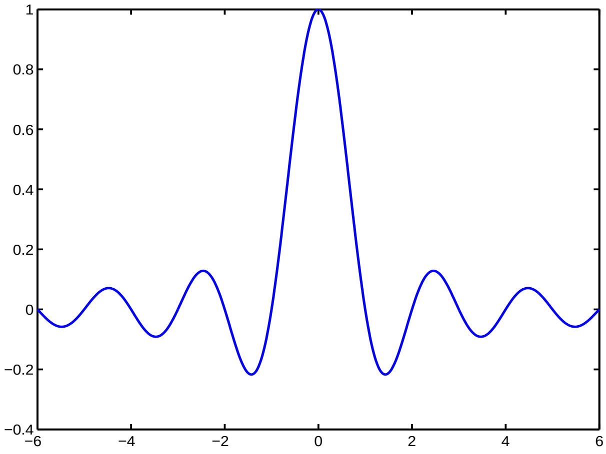

Figure 5.5. Scaling function ϕ of Sinc

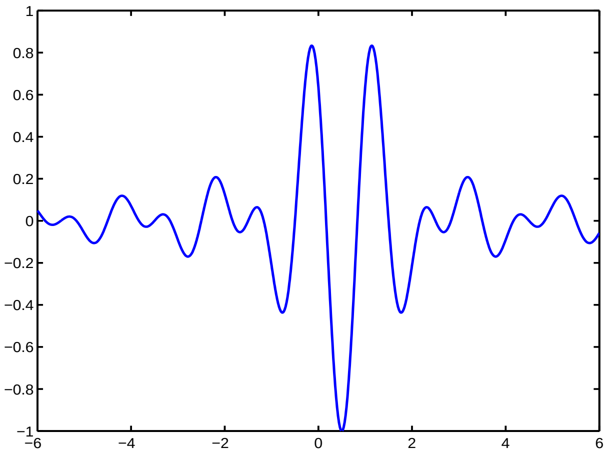

Figure 5.6. Wavelet function ψ of Sinc

considered for control problems that need function approximation.

Case 3. Mexican Wavelet

Mexican wavelet, described by $g(x) = (1 - x^2)e^{-\frac{x^2}{2}}$ and illustrated in Figure 5.7, is in fact a continuous wavelet. However we can see the desirable properties from the figure: very smooth and well localized. In practice we could use it as wavelet bases with appropriate discretization.

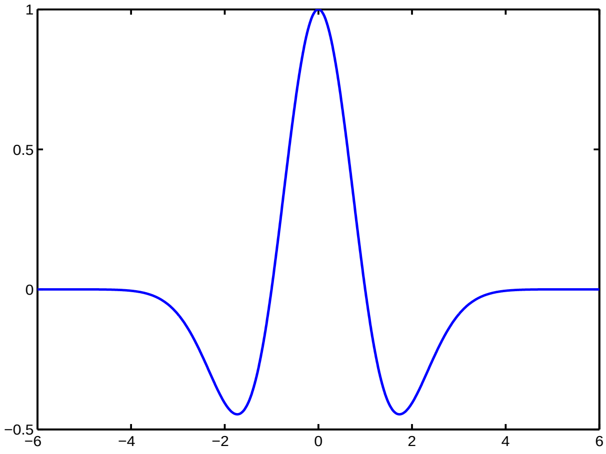

Figure 5.7. Mexican wavelet function $g(x)$

5.7 Illustrative Example

In order to provide useful information and guideline for practical applications of wavelets in ALC or RALC, we focus on a few important factors: the suitability of a wavelet basis, the complexity of the function approximation network, and the length of adaptive learning period. Let N_i and N_b denote the total number of iterations and the number of bases in the learning process respectively. Let N be the number of the iterations between the two structured updating, here N is the "dwell iterations", namely $k(i)$ increases by one when i increases by 10. Due to space limit, we will only demonstrate ALC and RALC for plants S_I and S_{II} under the initial condition a).

5.7.1 Adaptive Learning Control

Consider the following dynamic system

$$\begin{aligned} \dot{x}_1 &= x_2 \\ \dot{x}_2 &= 8e^{-x_1}\sin x_1 + u. \end{aligned} \tag{5.73}$$

The desired trajectory is $x_r(t) = t^3$, the augmented tracking error is $\sigma = \Delta x_1 + \Delta x_2$. The dynamic system is repeatable over $[0,\ 1]$.

Case 1. Orthonormal Wavelet db3

The orthonormal wavelets db3 is employed. The wavelet network structure is fixed at the resolution $j = 5$, a relatively finer resolution. The tracking error is shown in Figure 5.8. From the figure, we can see that the speed of convergence is rather slow although the structure is complex. Due to the lack of smoothness, db3 wavelet is not suitable for ALC.

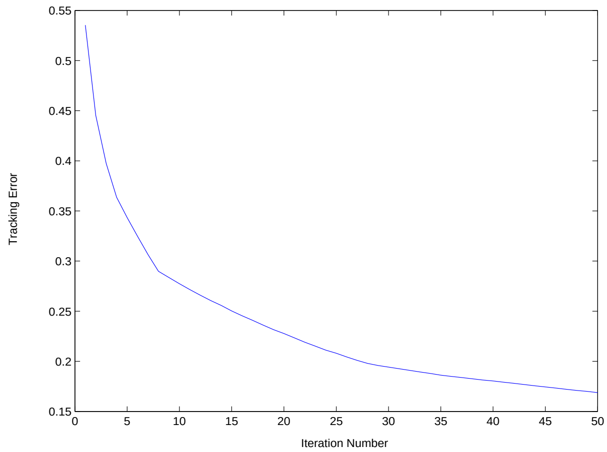

Figure 5.8. Tracking error with coarse structure $j = 5$.

Case 2. Sinc-wavelet

The error bound is set to be $\epsilon = 0.035$. First, the wavelet network structure is fixed at a coarse resolution $j = 0$. The tracking error is shown in Figure 5.9. From the figure, the tracking error is kept at a rather large level despite adaptive learning. This is due to the inadequate function approximation precision with the coarse resolution $j = 0$. Next we adjust the wavelet network structure by increasing one resolution when the iteration number i increase by one, that is, the dwell time iteration $N = 1$. The tracking error is shown in Figure 5.10. From the figure, the tracking error enters the pre-specified error bound after 7 iterations, indicating

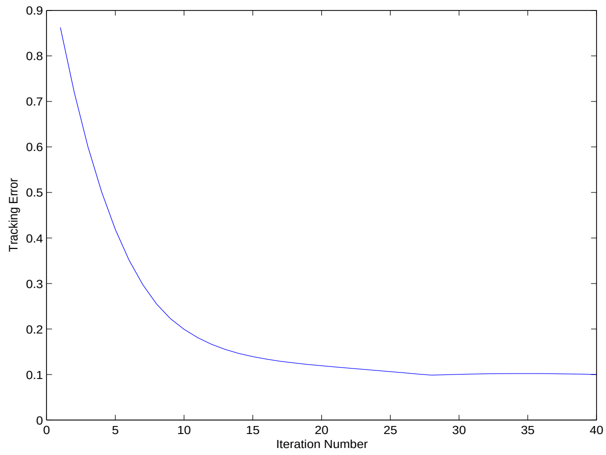

Figure 5.9. Tracking error at the resolution $j = 0$.

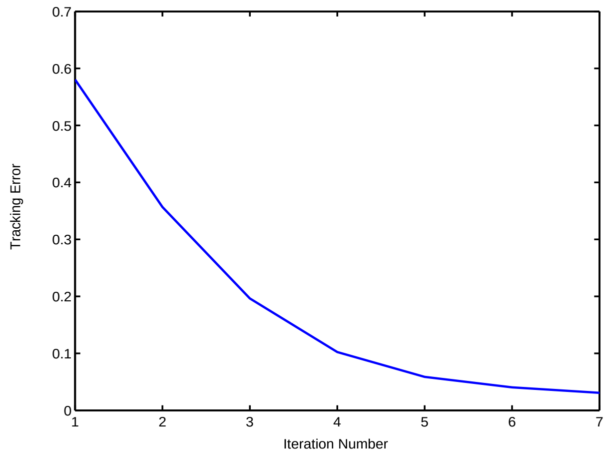

Figure 5.10. Tracking error when the resolution increases from 0 to 6 (Case 2)

a very fast convergence speed. This clearly shows the necessity to increase the number of bases.

On the other hand, resolution $j = 6$ corresponds to a relatively complex structure. A question arises: whether resolution $j = 6$ is really imperative? Note that updating the structure at every iteration, that is, $k(i) = i$ or $N = 1$, is the fastest updating speed. Since adaptive learning control needs time to reach steady state, we can update the network structure in a lower speed, for instance updating once after a few learning iterations. Choose different dwell iterations $N = 5$, $N = 10$ and $N = 15$, the comparison results are summarized in Table 5.1. From Table 5.1,

Table 5.1. Comparison for different dwell iterations

Dwell iterations	j	N_i
1	6	7
5	4	21
10	2	30
15	2	45

we can conclude that the resolution $j = 2$ is necessary and adequate. The tracking error for dwell iteration $N = 10$ is given in Figure 5.11. Table 5.1 indicates the cor-

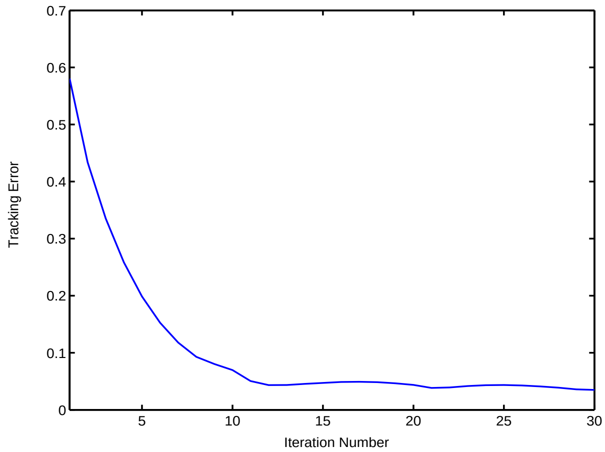

Figure 5.11. Tracking error with dwell iteration $N = 10$ (Case 2)

relation, or the trade-off between the learning speed and controller complexity. In practical control applications, the dwell iteration N can be determined according to other control requirements. For instance, if the priority is given to the learning speed, a small N would be proper. On the contrary, if the controller complexity is the main concern, a large N shall be chosen.

Case 3. Mexican Wavelet

Let the error bound be $\epsilon = 0.035$. Choose the dwell iteration $N = 1$, the tracking error is shown in Figure 5.12. It gives a better performance than Case 2 with sinc-wavelet. Next choose different dwell iterations $N = 5$ and $N = 10$, the

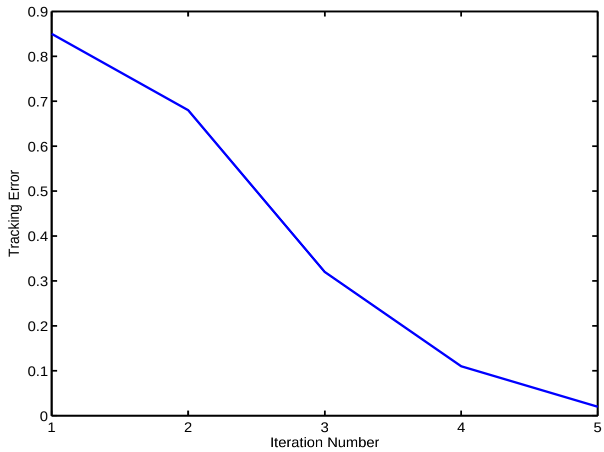

Figure 5.12. Tracking error by increasing j from 0 to 4 (Case 3)

comparison results are summarized in Table 5.2. From Table 5.2, it is obvious

Table 5.2. Comparison for different dwell iterations

Dwell iterations	j	N_i
1	4	5
5	2	11
10	1	14

that Mexican wavelet achieves a faster convergence speed and meanwhile uses a simpler structure. The tracking error with dwell iteration $N = 10$ is shown in Figure 5.13. The comparison studies show that Mexican wavelet is most suitable for control purpose.

5.7.2 Robust Adaptive Learning Control

Same as the preceding subsection, let the desired trajectory be $x_r(t) = t^3$, the augmented tracking error be $\sigma = \Delta x_1 + \Delta x_2$, and the dynamic system is repeatable over $[0, 1]$. The pre-specified tracking error bound is 0.01.

Case 1. RALC for Plant S_I

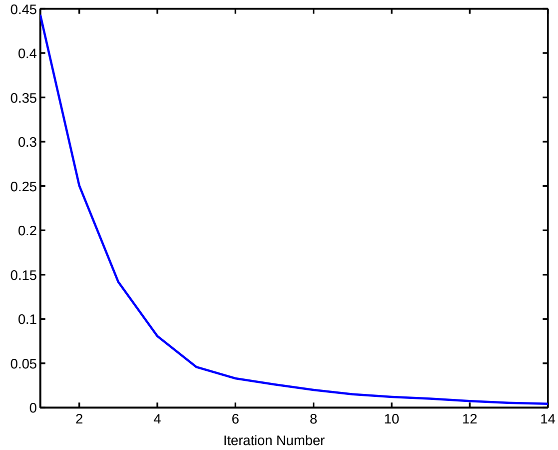

Figure 5.13. Tracking error with dwell iteration $N = 10$ (Case 3)

Consider the dynamic system

$$\begin{aligned}\dot{x}_1 &= x_2 \\ \dot{x}_2 &= 5x_2 \sin(x_1 + x_2) + u.\end{aligned}$$

The unknown nonlinear uncertainty $5\sin(x_1+x_2)x_2$ has an upper bounding function $5|x_2|$.

First choose different dwell iterations $N = 5$, $N = 10$ and $N = 15$, the comparison results are summarized in the Table 5.3. Here sinc-wavelet is used. From Table

Table 5.3. Comparison for different dwell iterations

Dwell iterations	j	N_i
5	2	14
10	1	20
15	1	24

5.3, satisfactory responses were achieved by RALC.

Next we investigate the effect of different initial resolutions. One of the practical control requirements is, whenever possible, to obtain the pre-specified tracking error using the minimum number of bases. Assume the scaling function is chosen at resolution $j = j_1$, which is the initial resolution. If the number of bases at j_1 layer is n_1, the total number of bases in the wavelet network at the resolution $j = j_n$ is

$$\sum_{j=j_1}^{j_n}[1 + I(2^{j+1}\frac{n_1 - 1}{2^{(-j_1+1)}})], \tag{5.74}$$

where the function $I(a)$ is equal to a when a is an integer number, or equal to an integer number nearest to a from above when a is not an integer number. The equation (5.74) shows that the number of bases is determined by three factors: the initial resolution j_1, the number of initial bases n_1, and the number of layers $j_n - j_1$. The number of bases increases rapidly if the initial resolution is chosen at a finer level, that is, with larger j_1. Therefore, in order to fully make use of the flexibility achieved by the network structural evolution, it is preferred to let the wavelet network start from a lower resolution j_1.

Choosing different initial resolutions $j_1 = -3$, $j_1 = -2$ and $j_1 = 0$, the comparison results are displayed in Table 5.4. Here Mexican-wavelet is used and the dwell iteration is $N = 10$. The minimum number of bases is $N_b = 92$, which is the case

Table 5.4. Comparisons for different initial resolutions

Initial resolution j_1	Final resolution j_n	N_b	N_i
-3	-1	92	25
-2	0	148	25
0	2	516	26

with the scaling function at resolution $j_1 = -3$. Note that the number of learning iterations are almost the same for three cases, hence there is no sacrifice of learning speed when the lowest resolution is used. In other words, $j_1 = -3$ achieves the best performance.

So far we only discussed the increment of a network, which may contain significant redundancies. Many network pruning algorithms have been proposed to reduce the neural network size. The simplest algorithm is to remove a node which is always with a very low weighting. By incorporating this simple algorithm, the wavelet network size can be further reduced to about one third. It was found that most wavelets nearby the boundary of $\mathcal{D}$ are not activated, implying that the actual state trajectory concentrates on only a portion of the compact set $\mathcal{D}$. The reduced number of bases is given in the following Table. Table 5.5 shows that the number

Table 5.5. Comparisons for different initial resolution

Initial resolution j_1	Final resolution j_n	N_b	N_i
-3	-1	28	29
-2	0	36	30
0	2	96	27

of bases is the minimum for the scaling function at the resolution $j_1 = -3$ and the number of iterations is again almost the same at different resolutions. Therefore, scaling function at resolution $j_1 = -3$ is optimal for this example.

Case 2. RALC for Plant S_{II}

Consider the following plant

$$\begin{aligned} \dot{x}_1 &= x_2 \\ \dot{x}_2 &= f(\mathbf{x}) + b(\mathbf{x})u, \end{aligned} \tag{5.75}$$

where $f(\mathbf{x}) = 5\sin(x_1 + x_2)x_2$ with the bounding function $5|x_2|$ and $b(\mathbf{x}) = (1 + |\sin(x_1)|)$ with the lower bound $b_0 = 1$. In this case, sinc-wavelet is chosen as the base wavelet. Choosing the dwell iteration $N = 15$, the tracking error is shown in Figure 5.14. This confirms the validity of the proposed robust adaptive learning control method for the dynamical system S_{II} with unknown input coefficient.

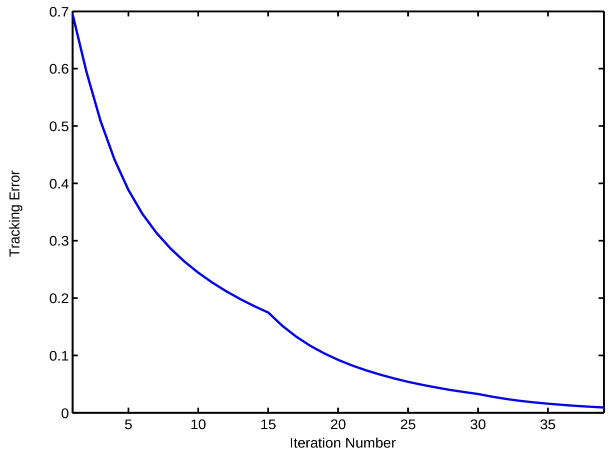

Figure 5.14. Tracking error with dwell iteration $N = 15$

5.8 Conclusion

In this chapter we developed an adaptive learning control approach which can fully make use of the powerful function approximation in a more flexible and constructive manner. The wavelet network provides an orthonormal basis for $\mathcal{L}^2(R)$ and can be constructed from the multiresolution approximation, thus can fulfill all requirements of the adaptive learning control approach. To concentrate on the idea, concepts and the basic methods, we only consider three classes of nonlinear uncertain dynamics: the simplest higher order plants with a lumped uncertain nonlinear function, plants with partially unknown input coefficient, and plants in cascade form. With rigorous analysis, we prove the existence of solution and learning convergence properties. A number of case studies are presented to demonstrate the effectiveness of wavelet based adaptive learning control, as well as the choice and design issues of wavelet network.

Chapter 6

On Initial Conditions in Iterative Learning Control

6.1 Introduction

Learning control enhances the system performance through repeated or cyclic operations. Iterative learning control deals with finite time interval tracking tasks that repeat, whereas repetitive learning control copes with periodic tracking tasks over infinite time interval.

To make a process convergent in a finite time interval, the initial condition becomes crucial because asymptotical convergence along the time horizon is no longer valid. Iterative learning control (ILC) based on contraction mapping requires the identical initial condition (*i.i.c.*) in order to achieve a perfect tracking (Arimoto *et al.*, 1984*b*; Sugie and Ono, 1991; Ahn *et al.*, 1993; Xu and Tan, 2003). The robustness of contraction based ILC has been studied (Arimoto *et al.*, 1991; Lee and Bien, 1991; Porter and Mohamed, 1991*b*; Porter and Mohamed, 1991*a*; Heinzinger *et al.*, 1992; Saab, 1994), and several algorithms were proposed for ILC without *i.i.c.* (Park and

Bien, 2000; Sun and Wang, 2002; Chen *et al.*, 1999).

Recently, new ILC approaches based on Lyapunov theory (Xu and Tan, 2003; Xu and Tan, 2002*a*; Qu, 2002; Jiang and Unbehauen, 2002; Tayebi, 2004) have been developed to complement the contraction mapping based ILC in the sense that local Lipschitz nonlinearities can be taken into consideration. Majority of those approaches also require the identical initial condition. In practical applications, the perfect initial resetting may not be obtainable. That motivates us to study initial conditions for this class of ILC.

In the chapter, five different initial conditions to be investigated are: a) identical initial condition (*i.i.c.*); b) progressive *i.i.c.*, i.e. the sequence of initial errors belong to l^2; c) fixed initial shift; d) random initial condition within a bound; e) alignment condition, i.e., the end state of the preceding iteration becomes the initial state of the current iteration.

Condition b) has not been exploited in contraction mapping based ILC. In the Lyapunov based ILC, this condition has been briefly mentioned in (French and Rogers, 2000*b*) wherein the unknowns are constant parameters. Hence, analogous to adaptive control, differential type adaptation law can be derived by the use of a quadratic Lyapunov function. In this chapter, we consider more general time-varying parametric uncertainties, wherein a difference type learning law is derived from a Lyapunov functional. A contribution of this chapter is to show the pointwise learning convergence under Condition b).

Condition c) has been studied in contraction mapping based ILC. In (Park and Bien, 2000), it shows that the tracking error can converge exponentially along the time axis from the fixed initial shift which cannot be eliminated. In (Sun and Wang, 2002), by rectifying the reference trajectory nearby the initial stage into a new one aligned with the actual initial value, the uniform convergence of the tracking error can be achieved. Condition c) has not been studied in Lyapunov based ILC. A contribution of this chapter is to demonstrate the similar learning performance: the tracking error will enter a designated bound with the fixed initial shift, and pointwisely converges when the reference trajectory can be rectified.

The effect of Condition d), which reflects the ILC robustness property, has been investigated in contraction mapping based ILC, e.g. (Heinzinger *et al.*, 1992) and (Park and Bien, 2000). The results show that the tracking error is confined to a bound which depends continuously on the bound of the initial state error. In a special case of Condition d), an initial state learning algorithm (Chen *et al.*, 1999) has been proposed to make the initial state a convergent sequence, subject to the maneuverability of the system initial states. By a rectifying action (Sun and Wang, 2002), the tracking error can also be confined to a finite bound which is proportional to the bound of the initial state error. As for the Lyapunov based ILC, the only report on Condition d) was given by (Jiang and Unbehauen, 2002), in which a switching control together with a reducing deadzone is used. In comparison, the contribution in this chapter is to show that the proposed ILC, which is a continuous control law, can converge to a designated bound under Condition d), or converge pointwisely when an appropriate rectifying action is taken.

Condition e) is not applicable in contraction mapping based ILC. In Lyapunov based ILC, our previous work (Xu, 2002*b*) has shown the learning convergence under Condition e). In this chapter, we first show that the learning convergence or boundedness with respect to conditions a-d) and e), though very different, can be easily discussed and determined under a unified framework using a Lyapunov functional. Next, under the same framework, the learning convergence speed can be evaluated for the conditions c), d) and e).

The objective of ILC is to achieve a convergent sequence in a function space. As such, the sequence approaches the desired one either in a pointwise manner, in L^p norm or in uniform norm. In the analysis of contraction based ILC, often the uniform norm is used. However, the uniform convergence is rather difficult to achieve in many control problems, especially for tracking tasks in a function space. In this chapter, we demonstrate that, a learning sequence can converge either pointwisely or in L^2 norm. L^2 norm is defined as $\|e_i\|_T \triangleq (\int_0^T e_i^2 dt)^{\frac{1}{2}}$.

The chapter is organized as follows. Section 6.2 states the problem and ILC algorithm. In Section 6.3, the learning convergence properties are analyzed under different initial conditions. Section 6.4 presents an illustrative example.

6.2 Problem Statement

Considering a tracking task that ends in a finite interval and repeats, ILC applies from iteration to iteration. To focus on the main theme with initial conditions, consider simple first order nonlinear dynamic system in the $i-$th iteration

$$\dot{x}_i = \theta(t)\xi(x_i, t) + u_i \qquad x(0) = x_0, \tag{6.1}$$

where $\xi(x_i, t)$ is a known nonlinear function which can be local Lipschitzian and the unknown time-varying parameter $\theta(t) \in \mathcal{C}[0,\ T]$. For notational convenience, in subsequent context we will omit the argument t for all variables and denote a function $\xi(x_i, t)$ as ξ_i where no confusion arises.

The reference trajectory is generated by a dynamics

$$\dot{x}_r = f(x_r, r, t), \tag{6.2}$$

where $f_r = f(x_r, r, t)$ is a known smooth function, r is a reference input which yields a bounded state $x_r(t)$ over the interval $[0,\ T]$. The tracking error is defined as $e_i(t) = x_r(t) - x_i(t)$.

The objective of ILC is to find a sequence of appropriate control input $u_i(t)$ for $t \in [0,\ T]$ such that the system state x_i tracks the reference trajectory x_r as $i \to \infty$.

From the theory of differential equation, the orbit of the nonlinear dynamics (6.1) is jointly determined by the initial value x_0 and the exogenous input u_i. A tiny discrepancy in initial conditions may lead to completely different orbits. However, a perfect initial resetting requires that the control system be equipped with a precise homing mechanism, which may not be possible for many practical engineering systems. Henceforth, the ultimate objective of this chapter is to relax this requirement with several less strict initial conditions, and investigate how does the learning performance alter accordingly. Consider the following five initial conditions:

a) $e_i(0) = 0$;

b) $\sum_{i=1}^{\infty} e_i^2(0) = C$, where C is a constant;

c) $|e_i(0)| = C \neq 0$, where C is a constant;

d) $e_i(0)$ is random and bounded by a constant C;

e) $e_i(0) = e_{i-1}(T)$.

Condition a) is the identical initial condition ($i.i.c.$) that is widely assumed for most ILC algorithms. Condition b) is the progressive $i.i.c.$, it shows that the sequence of $\{e_i(0)\}$ belongs to l^2, or $e_i(0) \to 0$ as $i \to \infty$. Condition c) is the fixed initial shift. Obviously, Condition a) is a special case of Condition b), and Conditions a-c) are special cases of Condition d). Generally speaking, it is adequate to consider Condition d) the worst case, if our concern is regarding the ILC robustness on initial shifts. Nonetheless, we can derive better and quantitative results on learning convergence with Conditions a-d), as we will show in this chapter.

Condition e) is the alignment condition, which is different from other initial conditions. The initial resetting condition in ILC usually implies both spatial resetting and temporal resetting. While time resetting is natural for a task to be finished and repeated over a finite period, the spatial resetting is however not an easy job and not so imperative. Note that it is the spatial resetting which gives rise to extra implementation difficulty. In quite a number of practical applications, the process will restart from where it stopped in previous trial. Therefore the end state of the preceding iteration becomes the initial state of the new iteration, i.e. $x_{i-1}(T) = x_i(0)$. As far as the reference trajectory is spatially closed, namely $x_r(0) = x_r(T)$, Condition e) holds for all iterations. The alignment condition removes the spatial resetting requirement.

The error dynamics at the i-th iteration can be expressed as

$$\dot{e}_i = f_r - \theta(t)\xi_i - u_i. \tag{6.3}$$

The learning control mechanism consists of the control law

$$u_i = ke_i + f_r - \hat{\theta}_i(t)\xi_i, \tag{6.4}$$

and the parametric learning law

$$\hat{\theta}_i(t) = \mathrm{proj}(\hat{\theta}_{i-1}(t)) - \xi_i e_i(t) \quad \hat{\theta}_{-1}(t) = 0, \tag{6.5}$$

where

$$\mathrm{proj}(\cdot) \triangleq \begin{cases} \cdot & |\cdot| \leq \theta^* \\ \mathrm{sign}(\cdot)\theta^* & |\cdot| > \theta^* \end{cases}$$

and θ^* is the projection bound which is sufficiently large such that $\theta^* \geq \sup_{t\in[0,\ T]} |\theta(t)|$. In practice, θ^* can be arbitrarily large but finite.

Substituting the learning control law (6.4) into the error dynamics (6.3) yields the closed-loop error dynamics

$$\dot{e}_i = -ke_i - \phi_i(t)\xi_i, \tag{6.6}$$

where $\phi_i(t) \triangleq \theta(t) - \hat{\theta}_i(t)$.

6.3 Learning Convergence Under Initial Conditions

First derive the boundedness of tracking error e_i and parameter estimate $\hat{\theta}_i$ under learning control law (6.4) and (6.5). Note that at the initial iteration $i = 0$, there is no parametric learning as $\hat{\theta}_{-1}(t) = 0$, and $\hat{\theta}_0 = -\xi_0 e_0(t)$. Hence we have to derive the boundedness of $(e_0, \hat{\theta}_0)$ in a way different from that for $(e_i, \hat{\theta}_i)$ with $i \geq 1$.

Proposition 6.1. *$(e_0, \hat{\theta}_0)$ is bounded for $t \in [0,\ T]$.*

Proof is given in Appendix A.3.

Now we can prove the boundedness of $(e_i, \hat{\theta}_i)$, which is summarized in the following theorem.

Theorem 6.1. *Under the initial conditions a)-d), the learning control law (6.4) and (6.5) ensures bounded $(e_i, \hat{\theta}_i)$ for any $i \geq 1$.*

Proof is given in Appendix A.4.

Since any two iterations are correlated via the learning law, the impact from an initial condition to the system performance could be in an accumulative fashion. The following proposition describes such an accumulative impact and facilitate subsequent analysis on the relationship between initial conditions and learning convergence.

Proposition 6.2. *The inequality*

$$\lim_{i\to\infty} V_i(t) \leq V_0(t) + \lim_{i\to\infty} \frac{1}{2}\sum_{j=1}^{i} e_j^2(0) - \lim_{i\to\infty}\sum_{j=1}^{i}\int_0^t k e_j^2 d\tau - \lim_{i\to\infty}\frac{1}{2}\sum_{j=1}^{i-1} e_j^2(t) \quad (6.7)$$

holds for $\forall\ i$, where V_i is a Lyapunov functional defined as

$$V_i(t) = \frac{1}{2}e_i^2(t) + \frac{1}{2}\int_0^t \phi_i^2(\tau)d\tau. \quad (6.8)$$

Proof is given in Appendix A.5.

Now we are in a position to demonstrate the main results summarized in Theorem 6.2. First, in addition to the boundedness of $(e_i, \hat{\theta}_i)$, we can achieve better learning performance under initial conditions a-d). Second, we are able to achieve L^2 learning convergence under the alignment condition e). Third, under the same framework with the Lyapunov functional, it is possible to further evaluate the learning convergence speed.

Theorem 6.2. *Part 1. Under the initial conditions a) and b), the tracking error e_i converges to zero pointwisely as $i \to \infty$;*
Part 2. Under the initial conditions c) and d), there exists a subsequence $\{e_{i_j}\}$ of $\{e_i\}$ such that for any arbitrary $\delta > 0$, $\|e_{i_j}\|_T \leq \epsilon$ as $i_j \to \infty$, where $\epsilon = \sqrt{\frac{C^2+\delta}{2k}}$.
Part 3. Under the alignment condition e), the tracking error $\|e_i\|_T$ converges to zero as $i \to \infty$.
Part 4. Under the conditions c) and d), for any given $\epsilon_0 > 0$ and $k > \frac{C^2}{2\epsilon_0^2}$, the tracking error $\|e_i\|_T$ will enter the ϵ_0-bound after at most $\frac{2V_0(T)}{2k\epsilon_0^2 - C^2}$ iterations. Furthermore, under the condition e), the tracking error $\|e_i\|_T \leq \epsilon_0$ after at most $\frac{2V_0(T)+e_1^2(0)}{2k\epsilon_0^2}$ iterations.

Proof: Part 1

First consider the initial condition a). With the condition, (6.7) is

$$\lim_{i\to\infty} V_i(t) \leq V_0(t) - \lim_{i\to\infty} \frac{1}{2}\sum_{j=1}^{i-1} e_j^2(t).$$

Consider the positiveness of V_i and boundedness of V_0, the sequence $e_i(t)$ converges to zero pointwisely as $i \to \infty$.

Next consider the initial condition b), $\sum_{i=1}^{\infty} e_i^2(0) = C$. The relation (6.7) becomes

$$\lim_{i\to\infty} V_i(t) \leq V_0(t) + \frac{1}{2}C - \lim_{i\to\infty} \frac{1}{2}\sum_{j=1}^{i-1} e_j^2(t).$$

The convergence property is analogous to a) because C is finite.

Part 2

The reduction to absurdity is applied. Suppose, on the contrary, there exists a positive integer N such that $\|e_i\|_T \geq \epsilon$ for all $i \geq N$.

Let $t = T$. The relation (6.7) with the initial conditions c) and d), $|e_i(0)| \leq C$, is

$$\begin{aligned}
\lim_{i\to\infty} V_i(T) &\leq V_0(T) + \lim_{i\to\infty}\frac{1}{2}iC^2 - \lim_{i\to\infty}\sum_{j=1}^{i}\int_0^T ke_j^2 d\tau - \lim_{i\to\infty}\frac{1}{2}\sum_{j=1}^{i-1}e_j^2(T) \\
&\leq V_0(T) + \frac{1}{2}NC^2 - \sum_{j=1}^{N}\int_0^T ke_j^2 d\tau \\
&\quad + \lim_{i\to\infty}\frac{1}{2}(i-N)C^2 - \lim_{i\to\infty}\sum_{j=N}^{i}\int_0^T ke_j^2 d\tau \\
&\leq B + \lim_{i\to\infty}\frac{1}{2}(i-N)C^2 - \lim_{i\to\infty}(i-N)k\epsilon^2 \\
&= B + \lim_{i\to\infty}(i-N)(\frac{1}{2}C^2 - k\epsilon^2) \qquad (6.9)
\end{aligned}$$

where

$$B = V_0(T) + \frac{1}{2}NC^2 - \sum_{j=1}^{N}\int_0^T ke_j^2 d\tau$$

is a finite constant. For arbitrary $\delta > 0$ and $\epsilon = \sqrt{\frac{C^2+\delta}{2k}}$, substitution into (6.9) we can obtain

$$\begin{aligned}
\lim_{i\to\infty} V_i(T) &\leq B + \lim_{i\to\infty}(i-N)(\frac{1}{2}C^2 - k\epsilon^2) \\
&\leq B - \lim_{i\to\infty}\frac{1}{2}(i-N)\delta \qquad (6.10)
\end{aligned}$$

The right hand side approaches $-\infty$ since B is finite, which leads to a contradiction with the fact that $V_i(T) \geq 0$.

Part 3

Let $t = T$ in (6.7). With the alignment condition e), $e_i(0) = e_{i-1}(T)$, we obtain the following relationship

$$\frac{1}{2}\sum_{j=1}^{i}e_j^2(0) - \frac{1}{2}\sum_{j=1}^{i-1}e_j^2(T) = \frac{1}{2}e_1^2(0),$$

and

$$\lim_{i\to\infty} V_i(T) \;\; \leq \;\; V_0(T) + \frac{1}{2}e_1^2(0) - \lim_{i\to\infty}\sum_{j=1}^{i}\int_0^T ke_j^2 d\tau.$$

Therefore

$$\lim_{i\to\infty}\int_0^T e_i^2 dt \triangleq \lim_{i\to\infty}\|e_i\|_T^2 = 0$$

because of the positiveness of V_i and the boundedness of V_0.

Part 4

Under the initial conditions c) and d), from (6.9) we have

$$\begin{aligned} V_i(T) &\leq V_0(T) + \frac{1}{2}iC^2 - \sum_{j=1}^{i}\int_0^T ke_j^2 d\tau - \frac{1}{2}\sum_{j=1}^{i-1}e_j^2(T) \\ &\leq V_0(T) + \frac{1}{2}iC^2 - \sum_{j=1}^{i}\int_0^T ke_j^2 d\tau \\ &= V_0(T) + \frac{1}{2}iC^2 - k\sum_{j=1}^{i}\|e_j\|_T^2. \end{aligned} \tag{6.11}$$

From (6.11), the larger the $\|e_j\|_T$, the faster the decease of $V_i(T)$. Let us assume a slowest decrease in $V_i(T)$, which corresponds to $\|e_j\|_T = \epsilon_0$ for all $j = 1, 2, \cdots, i$. Since

$$0 \leq V_0(T) + \frac{1}{2}iC^2 - k\sum_{j=1}^{i}\|e_j\|_T^2,$$

substituting $\|e_j\|_T = \epsilon_0$, we can derive $i \leq \frac{2V_0(T)}{2k\epsilon_0^2 - C^2}$ and $k > \frac{C^2}{2\epsilon_0^2}$.

Under the initial condition e), by observing the inequality

$$\begin{aligned} V_i(T) &\leq V_0(T) + \frac{1}{2}e_1^2(0) - \sum_{j=1}^{i}\int_0^T ke_j^2 d\tau \\ &= V_0(T) + \frac{1}{2}e_1^2(0) - k\sum_{j=1}^{i}\|e_j\|_T^2, \end{aligned}$$

the larger the $\|e_j\|_T$, the faster the decrease of $V_i(T)$. Similarly, substituting $\|e_j\|_T = \epsilon_0$ into the inequality

$$0 \leq V_0(T) + \frac{1}{2}e_1^2(0) - k\sum_{j=1}^{i}\|e_j\|_T^2,$$

we can obtain $i \leq \frac{2V_0(T)+e_1^2(0)}{2k\epsilon_0^2}$. □

Note that, in the Lyapunov based ILC, the state variables are accessible. A rectifying action can be taken to revise the reference trajectory such that its initial values are aligned with the actual ones. This leads to an improved learning performance for the initial conditions c) and d), as stated by the following corollary.

Corollary 6.1. *Let revised reference trajectory x_r^* be*

$$x_r^* = \begin{cases} x_r & \text{if} \quad t \in [h, T], \\ \tilde{x}_r & \text{if} \quad t \in [0, h), \end{cases} \tag{6.12}$$

where $h \in [0, T]$ can be chosen arbitrary and $\tilde{x}_r$ is a smooth function to link the initial position $x_i(0)$ and the reference trajectory $x_r(h)$ at the moment $t = h$. The less the h, the closer the revised reference trajectory to the original reference trajectory.

Obviously, $e_i(0) = 0$, i.e., initial condition a) is satisfied for the new reference trajectory. An interesting observation is, the tracking error dynamics (6.6) remains the same with respect to the new reference trajectory, even though the reference trajectory may vary at every iteration. Therefore, the pointwise convergence can be directly achieved in analogy to the result of initial condition a) in Theorem 6.2.

Remark 6.1. *From Part 3 of Theorem 6.2, a large gain k can reduce the tracking error bound ϵ under the initial condition c) and d). From Part 4 of Theorem 6.2, it can be seen that a large feedback gain k can also expedite the learning convergence speed.*

Remark 6.2. *The above results can be extended MIMO systems with multiple unknown parameters.*

Remark 6.3. *To speed up the parametric learning, a learning gain $\gamma > 0$ can be introduced in the parametric learning law*

$$\hat{\theta}_i = \hat{\theta}_{i-1} - \gamma \xi_i e_i.$$

Accordingly a factor γ^{-1} shall be multiplied to integral terms on the right hand side of Lyapunov functional, and the convergence analysis remains the same.

Remark 6.4. *It should be noted that in deriving the above convergence properties, we consider only sufficient conditions or the worst case performance. In practice, we may achieve better learning performance such as uniform convergence, although in theory only pointwise or L^2 convergence is guaranteed.*

6.4 Illustrative Example

Consider the system

$$\dot{x} = (1 + \sin \pi t)x^2 + u \qquad x(0) = x_0.$$

The reference model is $\dot{x}_r = -x_r + \sin^2 \pi t + 2$ with $x_r(0) = 1$. The tracking interval is $[0,\ 2]$. Throughout the simulation, choose the feedback gain $k = 1$ and parametric learning gain $\gamma = 1$. To measure the performance, we either calculate the sup-norm $|e_i|_{sup}$, i.e., the maximum tracking error of $|e_i(t)|$ over $[0,\ 2]$, or calculate L^2 norm $\| \cdot \|_{T=2}$.

Initial Condition a)

Let $e_i(0) = 0$, i.e., $x_i(0) = x_r(0) = 1$. The simulation result is shown in Figure 6.1. The learning convergence can be clearly seen.

Initial Condition b)

Let $e_i(0) = \frac{1}{i+1}$, then $C = \sum_{i=1}^{\infty} e_i^2(0) = (\frac{\pi^2}{8} + \frac{\pi^2}{6}) - 2$ is finite. The sup-norm of tracking error is displayed in Figure 6.2. It can be seen that the tracking error does converge, but not as fast as Condition a) due to the initial perturbations.

Initial Condition c)

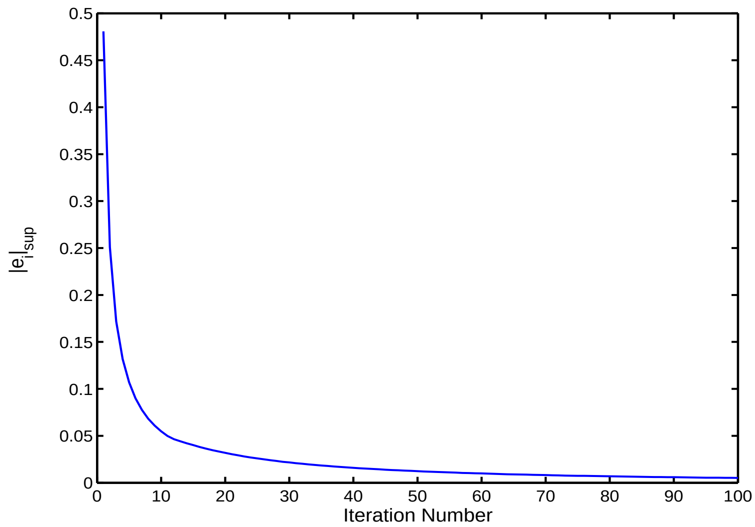

Figure 6.1. Learning convergence under initial condition a)

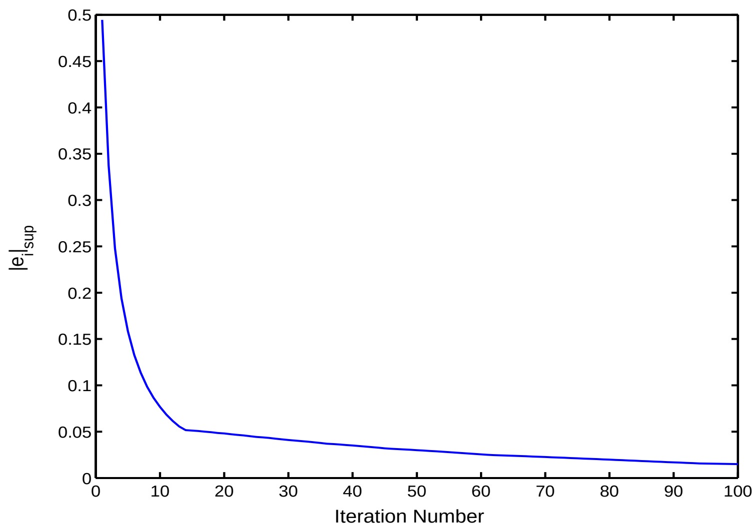

Figure 6.2. Learning convergence under initial condition b).

Consider a fixed initial shift $e_i(0) = -0.3$, namely $C = 0.3$. The theoretical tracking error bound is $\epsilon = \sqrt{C^2/2k} = 0.2121$. The tracking error profile is given in Figure 6.3. The tracking error can enter and stay well below the specified bound. In order to observe the effect of fixed initial shift, the tracking error profile at $100-$th iterations is shown in Figure 6.4. The control signal is given in Figure 6.5. In the time domain, it can be seen that the learning controller can quickly overcome the initial impact and converge to the reference trajectory. In the iteration domain, it

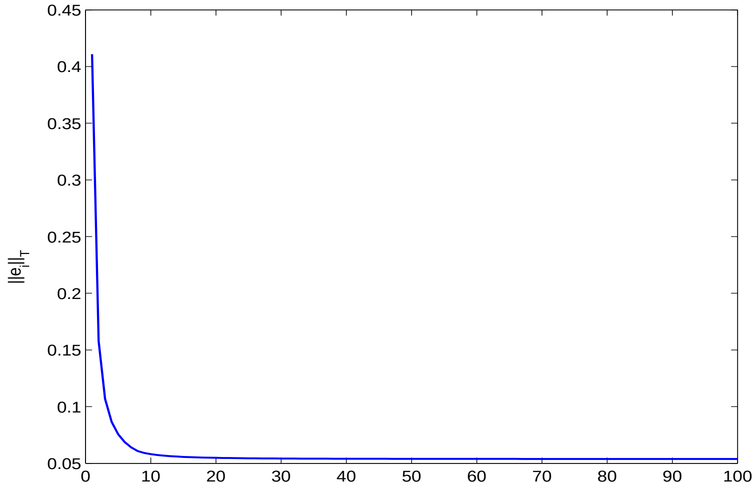

Figure 6.3. Learning convergence under initial condition c)

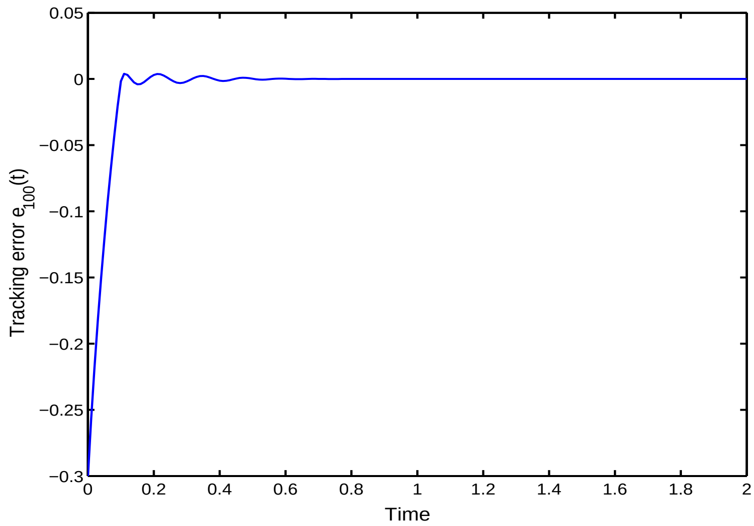

Figure 6.4. Tracking error at 100−th iterations under initial condition c)

can be seen that learning enters steady state after 10 iterations. Hence a simple stopping mechanism can be introduced in real applications: stop when the tracking error profile does not show significant reduction.

Initial Condition d)

Let $e_i(0)$ take values randomly in $[-0.3,\ 0]$. The bounded tracking performance is shown in Figure 6.6. The maximum error in each iteration is dominated by the

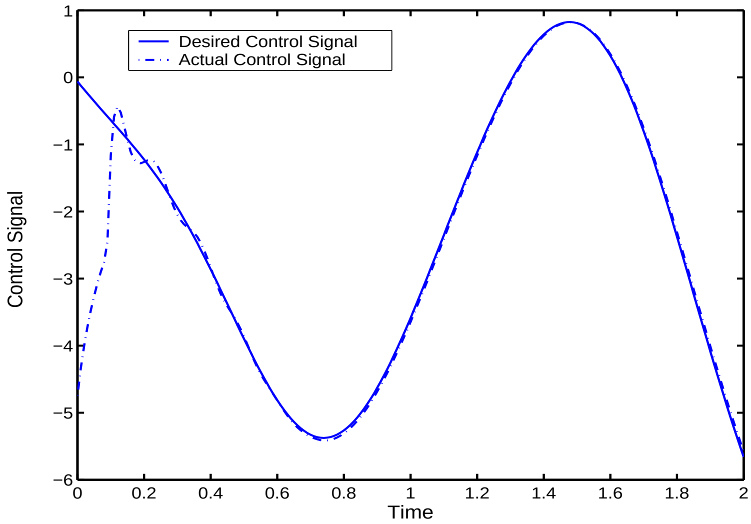

Figure 6.5. Control signal under initial condition c)

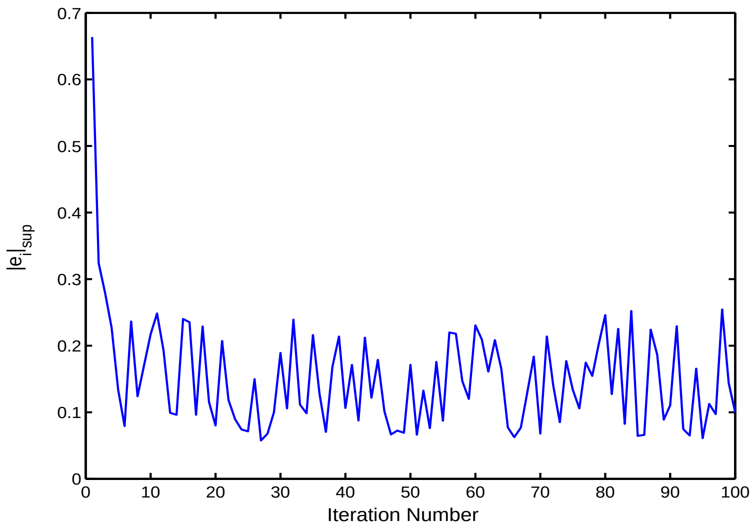

Figure 6.6. Bounded tracking performance under initial condition d)

initial error. The tracking error convergence is given in Figure 6.7. It can be seen that, despite the large initial error, the tracking error is kept at a much lower level for most time.

According to Corollary 6.1, the pointwise convergence of tracking error can be achieved if taking a rectifying action. In this example, for each iteration i, the

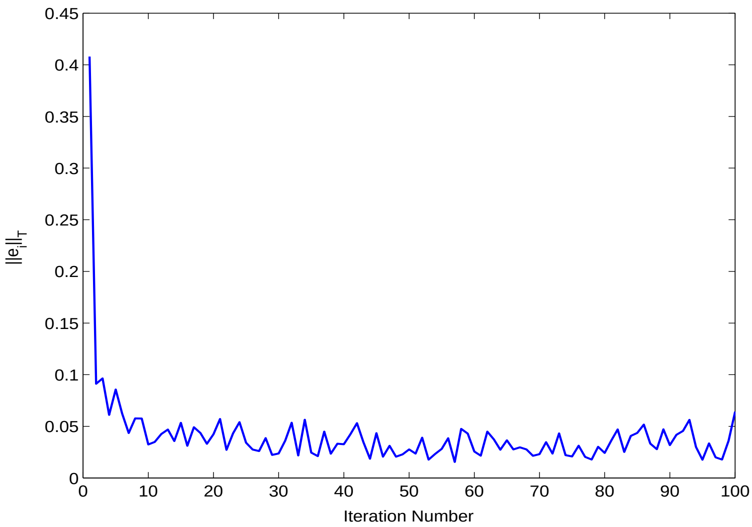

Figure 6.7. Learning convergence under initial condition d)

reference trajectory is revised as the following

$$x^*_{r,i} = \begin{cases} x_r, & \text{if} \quad t \in [h, T] \\ A_i t^2 + B_i t + C_i, & \text{if} \quad t \in [0, h) \end{cases}$$

where

$$A_i = \frac{\dot{x}_r(h)h + x_i(0) - x_r(h)}{h}, \qquad B_i = -\frac{\dot{x}_r(h)h + 2x_i(0) - 2x_r(h)}{h}, \qquad C_i = x_i(0).$$

Clearly, the revised reference trajectory remains the same in the time interval $[h, T]$. The coefficients of the quadratic function are chosen such that the revised portion $x^*_{r,i}(t)$ and its derivative are aligned with the original reference trajectory at $t = h$, meanwhile the revised reference trajectory is aligned with the initial state value at $t = 0$. Choose $h = 0.3$, the pointwise convergence of the tracking error is shown in Figure 6.8.

Initial Condition e)

Finally consider a spatially closed reference $x_r(t) = 1 - \cos(\pi t)$, i.e. $x_r(0) = x_r(2)$. Theoretically, in this case the tracking error only converges according to $\|\cdot\|_T$. Let $k = 3$ and $\gamma = 5$. The tracking error according to $\|\cdot\|_T$ norm is displayed in Figure 6.9. It validates the learning effect.

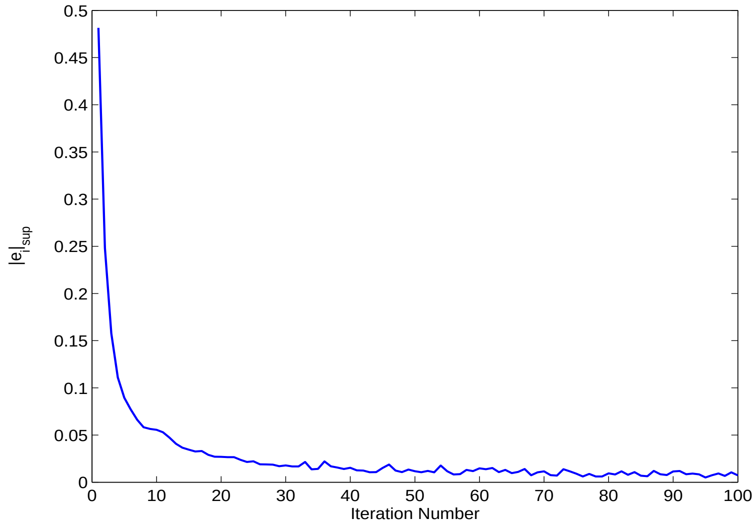

Figure 6.8. Pointwise convergence under initial condition d) by rectifying the reference trajectory

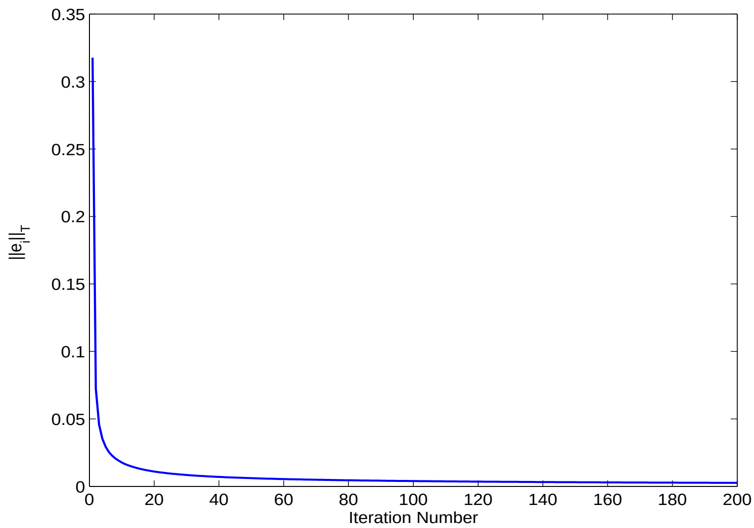

Figure 6.9. Learning convergence under initial condition e)

6.5 Conclusion

We discussed five different initial conditions associated with ILC. For each initial condition, the boundedness along the time horizon and asymptotical convergence along the iteration axis were exploited with rigorous analysis. Through both theoretical study and numerical examples, we can conclude that, the Lyapunov based

ILC can effectively work with sufficient robustness.

Chapter 7

Repetitive Learning Control for Nonlinear Systems with Parametric Uncertainties

7.1 Introduction

Learning control aims at improving the system performance via directly updating the control input, either repeatedly over a fixed finite time interval, or repetitively (cyclically) over an infinite time interval. Many learning control methods have been proposed in the past two decades, among them two predominant are iterative learning control (Arimoto *et al.*, 1984*a*), (Lee and Bien, 1997), (Moore *et al.*, 1992), (Chen and Wen, 1999), (Sun and Wang, 2001), (Chien and Yao, 2004) and (French and Phan, 2000) and repetitive control (Hara *et al.*, 1988), (Messner *et al.*, 1991) and (Longman, 2000), which can work effectively under repeatable control environment.

The repetitive control strategy has been widely applied in servo problems for LTI systems to track periodic references and reject periodic disturbances. The principal idea of the repetitive control, shown in Figure 7.1, is to embed a simple delay-based mechanism that updates the current cycle control input, $f(t)$, pointwisely by using the control input profile of the previous cycle, $f(t-T)$, and the output tracking error of the current cycle, $\sigma(t)$. It has been shown that (Nakano *et al.*, 1989), this simple delay-based mechanism plays the role as a universal internal model for all kinds of periodic references and/or periodic disturbances which are generated by LTI systems. It should be noted that the existing repetitive control is an input-output approach based on transfer functions. It requires the plant and all signal sources to be LTI, and the stability analysis is carried out in frequency domain using the small gain theorem. It achieves a geometric convergence speed over repetitions.

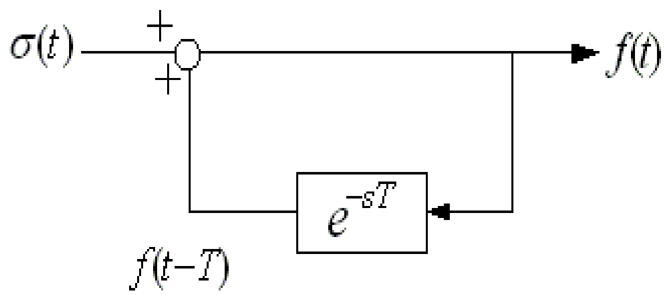

Figure 7.1. Repetitive learning mechanism

Under the present theoretical framework of repetitive control, however, it would be difficult to address the following two main issues.

A. Solving nonlinear servo problems which consist of two key-issues: A1) tracking a nonlinear reference model, either periodic or even non-periodic, and A2) dealing with plants with highly nonlinear components, such as local Lipschitz continuous functions.

B. Seeking a general control design in state space which also consists of two key-issues: B1) making full use of the system information regarding uncertainties and nonlinearities, and B2) using the well established Lyapunov theories to accomplish design with guaranteed asymptotic stability.

All four key issues above are inherently related. By making full use of the system information regarding uncertainties and nonlinearities (B1) in state space, the application of Lyapunov theories (B2) become possible. By using the well established Lyapunov theories (B2), it is possible to deal with nonlinear servo problems (A1) with highly nonlinear plants (A2). It will be shown in this chapter that, classifying the system uncertaities into parametric types (B1) will facilitate nonlinear servo design (A1). In particular it will be possible to track a non-periodic reference asymptotically.

In this chapter, our first objective is to establish a new control strategy – repetitive learning control (RLC) which, while retains the learning ability of the traditional repetitive control, directly addresses the above issues A and B. The new strategy is a direct extension of the recent advances in nonlinear learning control methods, including finite interval learning (Ham *et al.*, 2001) and (Xu and Tan, 2002*a*) which can be regarded as the generalization of the iterative learning control, and infinite interval learning (Dixon *et al.*, 2003) and (Cao and Xu, 2001) which can be regarded as the generalization of the repetitive control. Inheriting from repetitive control, the new control strategy will incorporate the simple delay-based loop into a nonlinear learning mechanism, hence be able to learn any periodic factors resulting from unknown but periodic parameters.

Note that the delay-based learning mechanism of RLC actually forms a continuous-time difference equation, and is of infinite dimensions. Considering the nonlinearities in the plant and control law, the repetitive learning control system is described by a set of mixed nonlinear differential and continuous-time difference equations.

Very few results were reported for this class of systems when the closed-loop stability, convergence and boudnedness are concerned, except for some local analysis result (Pepe and Verriest, 2003). When the existence of solution is concerned, the well established results hitherto were given by (Cruz and Hale, 1970) and (Hale and Pedro, 1977), which however focus on nonlinear dynamic systems satisfying a contractive mapping. Furthermore, the classical Lyapunov function based methods cannot be applied to obtain the convergence property.

Our second objective of this chapter, then, is to provide a rigorous and global analysis with regards to the existence of solution and learning convergence for the RLC system described by mixed differential and continuous-time difference equations. Such a rigorous analysis is indispensable when targeting at developing the learning control theories into a new control paradigm, analogous to what has been accomplished for adaptive control theories in the past four decades. To achieve this objective, the Lyapunov-Krasovskii functional is first employed to show the boundedness of states for any finite learning cycles. Then by means of the mathematical induction method the result is extended to the entire time horizon. Next, using the system smoothness property to convert the problem into a set of neutral functional differential equations (EL'SGOL'TS, 1964) we are able to conclude the existence of solution in the large. As a consequence of the above analysis we can further derive the learning convergence property.

Robustness or the insensitivity to small perturbations is a highly desired property when a control scheme is to be implemented. It is safe to say, the robustness is a landmark of the maturity or accomplishment for any control methodologies. In this chapter our third objective is to develop two robustifying modifications: the projection and damping for the learning mechanism. The projection scheme, similar to the one used in adaptive control, is applicable when the boundary information of unknown components are available. It guarantees the uniform convergence. On the other hand, the damping, in a sense analogous to the well known σ-modification in adaptive control, does not require the boundary information of unknown periodic components but what it can warrant is a bounded tracking performance. Different from the adaptive control which concerns only constant unknowns, here the unknown periodic components in the RLC system can be either time-varying parameters. Hence the problem solving will be more challenging.

This chapter is organized as follows. In Section 7.2, the repetitive learning control problem with parametric uncertainties is formulated first. Then the existence of solution and learning convergence properties are analyzed in Section 7.3. In Section 7.4, the robustification and extension to more general cases are discussed. Two illustrative examples are given in Section 7.5, and the conclusion is given in Section 7.6.

7.2 Problem Formulation

Consider the following uncertain nonlinear system

$$\begin{aligned} \dot{x}_j &= x_{j+1}, & j = 1, 2, \cdots, n-1, \\ \dot{x}_n &= \boldsymbol{\theta}^T(t)\boldsymbol{\xi}(t, \mathbf{x}) + u(t), & \mathbf{x}(0) = \mathbf{x}_0, \end{aligned} \tag{7.1}$$

where $\mathbf{x} = [x_1, x_2, \cdots, x_n]^T$ is a state vector, $\boldsymbol{\theta}(t) = [\theta_1(t), \theta_2(t), \cdots, \theta_m(t)]^T$ is an unknown parameter vector with rapidly time-varying coefficients and $\boldsymbol{\xi}(t, \mathbf{x}) = [\xi_1(t, \mathbf{x}), \xi_2(t, \mathbf{x}), \cdots, \xi_m(t, \mathbf{x})]^T$ is a regressor vector. $\boldsymbol{\xi}(t, x)$ consists of known nonlinear functions which can be local Lipschitzian and continuously differentiable with respect to (w.r.t.) the arguments $\mathbf{x}$ and t. In this chapter, we will consider repetitive learning control in the infinite time horizon under a repeatable control environment. Here the repeatable control environment is defined as below.

Assumption 7.1. *The unknown parameters* $\boldsymbol{\theta}(t) \in \mathcal{C}^1_{PT}([0, \infty); \mathcal{R}^m)$.

The target trajectory is generated by a reference model

$$\begin{aligned}\dot{x}_{r,j} &= x_{r,j+1}, \qquad j = 1, 2, \cdots, n-1, \\ \dot{x}_{r,n} &= s(t, \mathbf{x}_r, r), \qquad \mathbf{x}_r(0)\end{aligned} \tag{7.2}$$

where $\mathbf{x}_r = [x_{r,1}, x_{r,2}, \cdots, x_{r,n}]^T$, $s(\mathbf{x}_r, r, t)$ is a known smooth function w.r.t. all arguments, r is a constant reference input, and $\mathbf{x}_r(0)$ is a vector of the initial states. Denote $\Delta\mathbf{x} = \mathbf{x} - \mathbf{x}_r = [\Delta x_1, \Delta x_2, \cdots, \Delta x_n]^T$, the dynamics of the tracking error $\Delta\mathbf{x}(t)$ is

$$\Delta\dot{\mathbf{x}} = A\Delta\mathbf{x} + \mathbf{b}[\mathbf{c}\Delta\mathbf{x} + \boldsymbol{\theta}^T(t)\boldsymbol{\xi}(t, \mathbf{x}) + u(t) - s(\mathbf{x}_r, r, t)] \tag{7.3}$$

where $\mathbf{b} = [0\ 0\ \cdots\ 0\ 1]^T$, and $\mathbf{c} = [c_1, c_2, \cdots, c_{n-1}, 1]$ is chosen such that

$$A \triangleq \begin{bmatrix} 0 & 1 & 0 & \cdots & 0 & 0 \\ 0 & 0 & 1 & \cdots & 0 & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots & \\ 0 & 0 & 0 & \cdots & 0 & 1 \\ -c_1 & -c_2 & -c_3 & \cdots & -c_{n-1} & -1 \end{bmatrix}$$

is an asymptotically stable matrix. Based on Lyapunov stability theory for LTI systems, for a given positive definite matrix $Q \in R^{n\times n}$, there exits a unique positive definite matrix $P \in R^{n\times n}$ satisfying the following Lyapunov equation

$$A^T P + PA = -Q.$$

Let λ_Q be the minimum eigenvalue of the matrix Q, $-\mathbf{w}^T Q\mathbf{w} \leq -\lambda_Q\|\mathbf{w}\|^2$ holds for any $\mathbf{w} \in R^n$.

The ultimate control objective is to find an appropriate control input $u(t)$ such that the tracking error $\|\Delta\mathbf{x}(t)\|$ converges to zero as $t \to \infty$.

Consider the error dynamics (7.3), the learning control mechanism is constructed as follows.

$$u(t) = -\hat{\boldsymbol{\theta}}(t)^T\boldsymbol{\xi}(t, \mathbf{x}) + s(\mathbf{x}_r, r, t) - \mathbf{c}\Delta\mathbf{x}, \tag{7.4}$$

and the parametric updating law is

$$\begin{aligned}\hat{\boldsymbol{\theta}}(t) &= \hat{\boldsymbol{\theta}}(t-T) + k(t)\sigma(t)\boldsymbol{\xi}(t,\mathbf{x}), \qquad (7.5)\\ \hat{\boldsymbol{\theta}}(t) &= \mathbf{0}, \quad \forall t \in [-T,\ 0],\end{aligned}$$

where $\sigma(t) = \mathbf{b}^T P \Delta\mathbf{x}$,

$$k(t) = \begin{cases} 0, & -T \le t < 0, \\ k_1(t), & 0 \le t < T, \\ q, & t \ge T, \end{cases} \qquad (7.6)$$

where $q > 0$ is a constant, $k_1(t)$ is chosen to be monotone and smooth such that $k(t)$ is a smooth function on $[-T, \infty)$.

Proposition 7.1. *(Zheng* et al.*, 1991) Consider the following Cauchy problem*

$$\dot{\mathbf{x}} = \mathbf{f}(t,\mathbf{x}), \quad \mathbf{x}(t_0) = \mathbf{x}_0. \qquad (7.7)$$

Suppose that $\mathbf{f}(t,\mathbf{x})$ is continuous for $(t,\mathbf{x})$ in a region Ω, and satisfies the local Lipschitz condition with respect to $\mathbf{x}$. Then the solution of Cauchy problem (7.7) can be continued to the boundary, $\partial\Omega$, of Ω (possible ∞).

According to (Driver, 1965) and (EL'SGOL'TS, 1964) (Chapter 5, §12), we have the following proposition:

Proposition 7.2. *Consider the following differential difference equation of neutral type*

$$\dot{\mathbf{x}}(t) = \mathbf{f}(t, \mathbf{x}(t), \mathbf{x}(t-\tau), \dot{\mathbf{x}}(t-\tau)), \qquad t \ge t_0,$$

where the retardation τ is assumed constant. If the function $\mathbf{f}$ is continuous for the arguments, and the initial function $\mathbf{x}_0$ has a continuous derivation for $t_0 - \tau \le t \le t_0$, then the solution $\mathbf{x}$ exists in the neighborhood of the point $t = t_0$.

7.3 Existence of Solution and Convergence

Substituting the learning control law into the dynamics (7.3) yields the closed-loop error dynamics

$$\begin{aligned}\Delta\dot{\mathbf{x}} &= A\Delta\mathbf{x} + \mathbf{b}[\mathbf{c}\Delta\mathbf{x} + \boldsymbol{\theta}^T(t)\boldsymbol{\xi}(t,\mathbf{x}) + u(t) - s(\mathbf{x}_r, r, t)] \\ &= A\Delta\mathbf{x} + \mathbf{b}\boldsymbol{\phi}^T(t)\boldsymbol{\xi}(t,\Delta\mathbf{x}) \qquad (7.8)\end{aligned}$$

where $\boldsymbol{\phi}(t) = \boldsymbol{\theta}(t) - \hat{\boldsymbol{\theta}}(t)$. In above equation, $\mathbf{x}$ in $\boldsymbol{\xi}$ is replaced by $\Delta\mathbf{x} + \mathbf{x}_r(t)$ where $\mathbf{x}_r(t)$ as a function of t is not an independent argument. For notational convenience, $\boldsymbol{\xi}(t, \Delta\mathbf{x} + \mathbf{x}_r(t))$ is denoted by $\boldsymbol{\xi}(t, \Delta\mathbf{x})$. In subsequent context we further omit the argument t for all variables where no confusion arises, and denote $\boldsymbol{\xi}(t, \Delta\mathbf{x})$ by $\boldsymbol{\xi}$.

From the error dynamics (7.8) and the repetitive learning control law (7.4) and (7.5), we have

$$\begin{aligned}\Delta\dot{\mathbf{x}} &= \mathbf{f}(t, \Delta\mathbf{x}, \hat{\boldsymbol{\theta}}) \\ \hat{\boldsymbol{\theta}}(t) &= \hat{\boldsymbol{\theta}}(t-T) + k(t)\mathbf{b}^T P\Delta\mathbf{x}\boldsymbol{\xi}, \qquad (7.9)\end{aligned}$$

where

$$\mathbf{f}(t, \Delta\mathbf{x}, \hat{\boldsymbol{\theta}}) = A\Delta\mathbf{x} + \mathbf{b}(\boldsymbol{\theta}(t) - \hat{\boldsymbol{\theta}}(t))^T\boldsymbol{\xi}$$

Clearly, (7.9) consists of differential and continuous-time difference equations of neutral type.

Theorem 7.1. *For system (7.9) under Assumption 7.1, the learning control mechanism (7.4)-(7.6) ensures the existence of solution* $(\Delta\mathbf{x}, \hat{\boldsymbol{\theta}})$ *in* $[0, \infty)$ *and the asymptotical convergence*

$$\lim_{t\to\infty}\int_{t-T}^{t} \Delta\mathbf{x}^2(\tau)d\tau = 0.$$

Proof. Define the regions $\Omega_i \triangleq [(i-1)T, iT) \times R^n, i = 1, 2, \cdots$, for $(t, \Delta\mathbf{x})$. The theorem proof consists of three parts. *Part 1* and *Part 2* prove the existence of solution in the intervals $[0, T)$ and $[T, \infty)$ respectively. *Part 3* derives the convergence of the tracking error $\Delta\mathbf{x}$.

Part 1. Existence of the solution $(\mathbf{x}, \hat{\boldsymbol{\theta}})$ *in* $[0, T)$

Firstly, we claim that the solution $(\Delta\mathbf{x}, \hat{\boldsymbol{\theta}})$ of the differential difference equation (7.9) exists in $[0, T)$. For $i = 1$, we have $\hat{\boldsymbol{\theta}}(t) = \mathbf{0}$ for $t \in [-T, 0]$. Therefore, by substituting $\boldsymbol{\theta}(t)$ into $\mathbf{f}$ the dynamics (7.9) renders to a set of ODE (Ordinary Differential Equation), and $\mathbf{f}(t, \Delta\mathbf{x}, \hat{\boldsymbol{\theta}}) : \Omega_1 \to R^n$ is continuous in $\Delta\mathbf{x}$ by virtue of the smoothness of $\boldsymbol{\xi}$. By Peano's Existence Theorem (Zheng *et al.*, 1991), associated with the initial condition $\Delta\mathbf{x}(0)$, the equation (7.9) has a continuous solution in a neighborhood of $t = 0$. Furthermore it is easy to check that $\mathbf{f}(t, \Delta\mathbf{x}, \hat{\boldsymbol{\theta}})$ is locally Lipschitzian in $\Delta\mathbf{x}$. We need only to consider the solution for $t > 0$. Assume $[0, t_1)$ be the maximal interval to which the solution $\Delta\mathbf{x}$ can be continued up. Proposition 7.1 implies that $\Delta\mathbf{x}$ tends to the boundary $\partial\Omega_1$ of Ω_1 as $t \to t_1$. It further implies that $\lim_{t\to t_1} \|\Delta\mathbf{x}\| = \infty$ if $t_1 \le T$, i.e., for any $C > 0$, there exists $\delta_1 > 0$ such that $\|\Delta\mathbf{x}\| \ge C$ for all $t \ge t_1 - \delta_1$. Since $\Delta\mathbf{x}$ exists for all $t \in [0, t_1 - \frac{\delta_1}{2}]$, define the following Lyapunov-Krasovskii functional:

$$V(t, \Delta\mathbf{x}, \boldsymbol{\phi}) = \frac{1}{2}\Delta\mathbf{x}^T P \Delta\mathbf{x} + \frac{1}{2q}\int_{t-T}^{t} \boldsymbol{\phi}^T(\tau)\boldsymbol{\phi}(\tau)d\tau.$$

Now we prove the finiteness of $V(t, \Delta\mathbf{x}, \boldsymbol{\phi})$ for all $t \in [0, t_1 - \frac{\delta_1}{2}]$. From the existence theorem of differential equation (Yoshizawa, 1975) there exists a $T_1 > 0$ and $[0, T_1) \subset [0, t_1 - \frac{\delta_1}{2}]$, the boundedness of $V(t, \Delta\mathbf{x}, \boldsymbol{\phi})$ over $[0, T_1)$ can be guaranteed and we need only focus on the interval $[T_1, t_1 - \frac{\delta_1}{2}]$. For any $t \in [T_1, t_1 - \frac{\delta_1}{2}]$, the upper right hand derivative of V is

$$\begin{aligned}\dot{V} &= \frac{1}{2}(\Delta\dot{\mathbf{x}}^T P\Delta\mathbf{x} + \Delta\mathbf{x}^T P\Delta\dot{\mathbf{x}}) \\ &\quad + \frac{1}{2q}[(\boldsymbol{\theta} - \hat{\boldsymbol{\theta}})^T(\boldsymbol{\theta} - \hat{\boldsymbol{\theta}}) - (\boldsymbol{\theta}_\ominus - \hat{\boldsymbol{\theta}}_\ominus)^T(\boldsymbol{\theta}_\ominus - \hat{\boldsymbol{\theta}}_\ominus)],\end{aligned}$$

where $\boldsymbol{\theta}_\ominus = \boldsymbol{\theta}(t-T)$ and $\hat{\boldsymbol{\theta}}_\ominus = \hat{\boldsymbol{\theta}}(t-T)$.

Substituting the dynamics (7.8), then

$$\begin{aligned}&\frac{1}{2}(\Delta\dot{\mathbf{x}}^T P\Delta\mathbf{x} + \Delta\mathbf{x}^T P\Delta\dot{\mathbf{x}}) \\ &= \frac{1}{2}[\Delta\mathbf{x}^T A^T + (\mathbf{b}\boldsymbol{\phi}^T\boldsymbol{\xi})^T]P\Delta\mathbf{x} + \frac{1}{2}\Delta\mathbf{x}^T P(A\Delta\mathbf{x} + \mathbf{b}\boldsymbol{\phi}^T\boldsymbol{\xi}) \\ &= \frac{1}{2}\Delta\mathbf{x}^T(A^T P + PA)\Delta\mathbf{x} + \mathbf{b}^T P\Delta\mathbf{x}\boldsymbol{\phi}^T\boldsymbol{\xi} \\ &\le -\frac{\lambda_Q}{2}\|\Delta\mathbf{x}\|^2 + \sigma\boldsymbol{\phi}^T\boldsymbol{\xi}. \end{aligned} \tag{7.10}$$

From the updating law (7.5), we have $\hat{\boldsymbol{\theta}}(t-T) = \mathbf{0}$ for all $t \in [0,\ T]$ and $\hat{\boldsymbol{\theta}}(t) = k_1(t)\sigma(t)\boldsymbol{\xi}$. Since $k_1(t)$ is strictly increasing in $[0,T]$, $\frac{1}{k_1(t)} \ge \frac{1}{q}$ is ensured in the time interval $[T_1, t_1 - \frac{\delta_1}{2}]$. We can derive

$$\begin{aligned}&\frac{1}{2q}(\hat{\boldsymbol{\theta}} - \boldsymbol{\theta})^T(\hat{\boldsymbol{\theta}} - \boldsymbol{\theta}) - \frac{1}{2q}\boldsymbol{\theta}_\ominus^T\boldsymbol{\theta}_\ominus \\ &\le \frac{1}{2k_1(t)}(\hat{\boldsymbol{\theta}} - \boldsymbol{\theta})^T(\hat{\boldsymbol{\theta}} - \boldsymbol{\theta}) - \frac{1}{2q}\boldsymbol{\theta}_\ominus^T\boldsymbol{\theta}_\ominus \\ &= \frac{\boldsymbol{\theta}^T\boldsymbol{\theta}}{2k_1(t)} - \frac{1}{k_1(t)}\hat{\boldsymbol{\theta}}^T(\boldsymbol{\theta} - \hat{\boldsymbol{\theta}}) - \frac{\hat{\boldsymbol{\theta}}^T\hat{\boldsymbol{\theta}}}{2k_1(t)} - \frac{1}{2q}\boldsymbol{\theta}_\ominus^T\boldsymbol{\theta}_\ominus \\ &= \frac{\boldsymbol{\theta}^T\boldsymbol{\theta}}{2k_1(t)} - \sigma\boldsymbol{\phi}^T\boldsymbol{\xi} - \frac{\hat{\boldsymbol{\theta}}^T\hat{\boldsymbol{\theta}}}{2k_1(t)} - \frac{1}{2q}\boldsymbol{\theta}_\ominus^T\boldsymbol{\theta}_\ominus,\end{aligned}$$

and $\dot{V}$ becomes

$$\begin{aligned}\dot{V} &= -\frac{\lambda_Q}{2}\|\Delta\mathbf{x}\|^2 + \sigma\boldsymbol{\phi}^T\boldsymbol{\xi} + \frac{\boldsymbol{\theta}^T\boldsymbol{\theta}}{2k_1(t)} - \sigma\boldsymbol{\phi}^T\boldsymbol{\xi} - \frac{\hat{\boldsymbol{\theta}}^T\hat{\boldsymbol{\theta}}}{2k_1(t)} - \frac{1}{2q}\boldsymbol{\theta}_\ominus^T\boldsymbol{\theta}_\ominus \\ &\le \frac{\boldsymbol{\theta}^T\boldsymbol{\theta}}{2k_1(t)}\end{aligned} \tag{7.11}$$

Integrating (7.11) from T_1 to t, we obtain

$$V(t, \Delta\mathbf{x}, \boldsymbol{\phi}) \le V(T_1, \Delta\mathbf{x}(T_1), \boldsymbol{\phi}(T_1)) + \frac{1}{2q}\int_{T_1}^{t}\frac{\boldsymbol{\phi}^T(\tau)\boldsymbol{\theta}(\tau)}{k_1(\tau)}d\tau.$$

Since $\boldsymbol{\theta}(t) \in \mathcal{C}^1_{PT}([0,\infty);\mathcal{R}^m)$, $\int_{T_1}^{t} \frac{\boldsymbol{\theta}^T(\tau)\boldsymbol{\theta}(\tau)}{k_1(\tau)} d\tau$ is bounded. Thus V is bounded for all $t \in [0, t_1 - \frac{\delta_1}{2}]$. Let $N^2\lambda_P > 0$ be the bound of V on $[0, t_1 - \frac{\delta_1}{2}]$, where λ_P is the minimum eigenvalue of the positive definite matrix P. Then N does not depend on δ_1. By the definition of Lyapunov functional V, we can see that $\|\Delta\mathbf{x}\| \leq \sqrt{V/\lambda_P} = N$ for all $t \in [0, t_1 - \frac{\delta_1}{2}]$. Taking $C = 2N$ in advance, for the corresponding $\delta_1 > 0$ we have

$$C \leq \|\Delta\mathbf{x}(t_1 - \frac{\delta_1}{2})\| \leq N = \frac{C}{2},$$

a contradiction which implies $t_1 \geq T$. This assures the solution $\Delta\mathbf{x}$ of the dynamic system (7.9) exists in $[0, T]$. Further, considering the smoothness of the right hand side of equation (7.9), $\Delta\mathbf{x}(t)$ and $\hat{\boldsymbol{\theta}}(t)$ are both continuously differentiable for any $t \in [0, T)$.

Part 2. Existence of the solution $(\Delta\mathbf{x}, \hat{\boldsymbol{\theta}})$ *in* $[T, \infty)$

Assume that the solution $(\Delta\mathbf{x}, \hat{\boldsymbol{\theta}})$ of the differential difference equation (7.9) exists in $[(j-1)T, jT)$ for $j = 2, \cdots, i-1$. This implies $\Delta\mathbf{x}$ and $\hat{\boldsymbol{\theta}}(t)$ are both continuously differentiable for $t \in [0, (i-1)T)$. Assume that the solution of (7.9) can be continued up to a time $t \in [(i-1)T, iT)$, by differentiating $\hat{\boldsymbol{\theta}}(t)$ we obtain

$$\begin{aligned} \Delta\dot{\mathbf{x}} &= \mathbf{f}(t, \Delta\mathbf{x}, \hat{\boldsymbol{\theta}}) \\ \dot{\hat{\boldsymbol{\theta}}}(t) &= \mathbf{g}(t, \mathbf{x}, \hat{\boldsymbol{\theta}}(t), \dot{\hat{\boldsymbol{\theta}}}(t-T)), \quad t \in [(i-1)T, iT), \end{aligned} \tag{7.12}$$

where

$$\begin{aligned} \mathbf{g}(t, \mathbf{x}, \hat{\boldsymbol{\theta}}(t), \dot{\hat{\boldsymbol{\theta}}}(t-T)) &= \dot{\hat{\boldsymbol{\theta}}}(t-T) + q\mathbf{b}^T P\mathbf{f}(t, \Delta\mathbf{x}, \hat{\boldsymbol{\theta}})\boldsymbol{\xi} \\ &\quad + q\mathbf{b}^T P\Delta\mathbf{x}\boldsymbol{\xi}_t + q\mathbf{b}^T P\Delta\mathbf{x}\Xi_x\mathbf{f}(t, \Delta\mathbf{x}, \hat{\boldsymbol{\theta}}), \end{aligned}$$

with $\boldsymbol{\xi}_t = \frac{\partial\boldsymbol{\xi}}{\partial t}$, and $\Xi_x = \frac{\partial\boldsymbol{\xi}}{\partial\mathbf{x}}$. Since the function $\mathbf{f}(t, \Delta\mathbf{x}, \hat{\boldsymbol{\theta}})$ and $\mathbf{g}(t, \Delta\mathbf{x}, \hat{\boldsymbol{\theta}}(t), \dot{\hat{\boldsymbol{\theta}}}(t-T))$ are continuous with respect to the arguments, and functions $\Delta\mathbf{x}$ and $\hat{\boldsymbol{\theta}}(t)$ have continuous derivatives on $[(i-2)T, (i-1)T)$. According to Proposition 7.2, the solution $(\Delta\mathbf{x}, \hat{\boldsymbol{\theta}})$ of the equation (7.12) exists at the neighborhood of the point $(i-1)T$. Furthermore, $\mathbf{f}(t, \Delta\mathbf{x}, \hat{\boldsymbol{\theta}}) : \Omega_i \to R^n$ is continuous and locally Lipschitzian in $\Delta\mathbf{x}$ and $\hat{\boldsymbol{\theta}}$. Thus the solution $\Delta\mathbf{x}$ can be continued up to the boundary $\partial\Omega_i$ of Ω_i. Let $[(i-1)T, t_i)$ be the maximal interval to which the solution $\Delta\mathbf{x}$ can be continued up. If $t_i \leq iT$, there exists a $\delta_i > 0$ such that $\|\Delta\mathbf{x}\| \geq C$ for all $t \geq t_i - \delta_i$. For $t \in [(i-1)T, t_i - \frac{\delta_i}{2})$, define the Lyapunov-Krasovskii functional

$$V(t, \Delta\mathbf{x}, \boldsymbol{\phi}) = \frac{1}{2}\Delta\mathbf{x}^T P \Delta\mathbf{x} + \frac{1}{2q}\int_{t-T}^{t} \boldsymbol{\phi}^T(\tau)\boldsymbol{\phi}(\tau) d\tau. \tag{7.13}$$

Then the upper right hand derivative of V is

$$\dot{V} = \frac{1}{2}(\Delta\dot{\mathbf{x}}^T P \Delta\mathbf{x} + \Delta\mathbf{x}^T P \Delta\dot{\mathbf{x}}) + \frac{1}{2q}(\boldsymbol{\phi}^T\boldsymbol{\phi} - \boldsymbol{\phi}_\ominus^T\boldsymbol{\phi}_\ominus). \tag{7.14}$$

Substituting the error dynamics (7.8) into the above equation, analogous to the relation (7.10) the first term on the right hand side is

$$\frac{1}{2}(\Delta\dot{\mathbf{x}}^T P \Delta\mathbf{x} + \Delta\mathbf{x}^T P \Delta\dot{\mathbf{x}}) \leq -\frac{\lambda_Q}{2}\|\Delta\mathbf{x}\|^2 + \sigma\boldsymbol{\phi}^T\boldsymbol{\xi}. \tag{7.15}$$

Now let us derive the second term on the right hand side of (7.14). Using the parametric learning law (7.5), the periodic property $\boldsymbol{\theta} = \boldsymbol{\theta}_\ominus$, and the algebraic relationship

$$(\mathbf{a}-\mathbf{b})^T(\mathbf{a}-\mathbf{b}) - (\mathbf{a}-\mathbf{c})^T(\mathbf{a}-\mathbf{c}) = -2(\mathbf{a}-\mathbf{b})^T(\mathbf{b}-\mathbf{c}) - (\mathbf{b}-\mathbf{c})^T(\mathbf{b}-\mathbf{c}) \tag{7.16}$$

where $\mathbf{a}$, $\mathbf{b}$, $\mathbf{c}$ are vectors with the same dimensions, we have

$$\begin{aligned}
\frac{1}{2q}(\boldsymbol{\phi}^T\boldsymbol{\phi} - \boldsymbol{\phi}_\ominus^T\boldsymbol{\phi}_\ominus) &= \frac{1}{2q}[(\boldsymbol{\theta}-\hat{\boldsymbol{\theta}})^T(\boldsymbol{\theta}-\hat{\boldsymbol{\theta}}) - (\boldsymbol{\theta}-\hat{\boldsymbol{\theta}}_\ominus)^T(\boldsymbol{\theta}-\hat{\boldsymbol{\theta}}_\ominus)] \\
&= \frac{1}{2q}[-2(\boldsymbol{\theta}-\hat{\boldsymbol{\theta}})^T(\hat{\boldsymbol{\theta}}-\hat{\boldsymbol{\theta}}_\ominus) - (\hat{\boldsymbol{\theta}}-\hat{\boldsymbol{\theta}}_\ominus)^T(\hat{\boldsymbol{\theta}}-\hat{\boldsymbol{\theta}}_\ominus)] \\
&= -\sigma\boldsymbol{\phi}^T\boldsymbol{\xi} - \frac{q}{2}\sigma^2\boldsymbol{\xi}^T\boldsymbol{\xi}. \qquad (7.17)
\end{aligned}$$

Substituting (7.15) and (7.17) into (7.14), the upper right hand derivative of V is

$$\dot{V} = -\frac{\lambda_Q}{2}\|\Delta\mathbf{x}\|^2 - \frac{q}{2}\sigma^2\boldsymbol{\xi}^T\boldsymbol{\xi} \leq -\frac{\lambda_Q}{2}\|\Delta\mathbf{x}\|^2. \tag{7.18}$$

Clearly $V(t, \Delta\mathbf{x}, \boldsymbol{\phi})$ will be bounded for $t \in [(i-1)T,\ t_i - \frac{1}{2}\delta_i)$ as far as $V(\tau, \Delta\mathbf{x}(\tau), \boldsymbol{\phi}(\tau))$ is bounded for $\tau \in [0,\ (i-1)T)$. Let $N^2\lambda_P$ be the bound of V on $[(i-1)T, t_i - \frac{\delta_i}{2})$, then N does not depend on δ_i. By the definition of Lyapunov-Krasovskii functional, we have $\|\Delta\mathbf{x}(t)\| \leq \sqrt{V/\lambda_P} = N$ for all $t \in [(i-1)T,\ t_i)$. Taking $C = 2N$ in advance, if the solution can only be continued up to $t_i < iT$, then we again has the contradiction

$$C \leq \|\Delta\mathbf{x}(t_i - \frac{\delta_i}{2})\| \leq N = \frac{C}{2}.$$

According to the theory of mathematical induction, the solution $\Delta\mathbf{x}$ exists in $t \in [(i-1)T,\ iT)$ for any finite i. Furthermore, since the solution $\hat{\boldsymbol{\theta}}(t)$ exists for $t \in [0,\ (i-1)T)$, then from

$$\hat{\boldsymbol{\theta}}(t) = \hat{\boldsymbol{\theta}}(t-T) + q\sigma(t)\boldsymbol{\xi}$$

and the existence of $\Delta\mathbf{x}$ for $t \in [(i-1)T, iT)$, the solution $\hat{\boldsymbol{\theta}}(t)$ exists for $t \in [(i-1)T,\ iT)$. Thus the solution $\Delta\mathbf{x}$ and $\hat{\boldsymbol{\theta}}(t)$ exists in $[0,\ iT)$ for any finite i. This implies that the solution $(\Delta\mathbf{x},\ \hat{\boldsymbol{\theta}})$ either is uniformly bounded or tends to infinity as $t \to \infty$. Thus $\Delta\mathbf{x}$ and $\hat{\boldsymbol{\theta}}(t)$ exist for $t \in [0, \infty)$.

Part 3. Asymptotical Convergence

Now derive the integral convergence

$$\lim_{t\to\infty} \int_{t-T}^{t} \|\Delta\mathbf{x}(\tau)\|^2 d\tau = 0$$

using the relation (7.18), that is, $\dot{V}$ is negative semi-definite for $t \in [T, \infty)$. Suppose that

$$\lim_{t\to\infty} \int_{t-T}^{t} \|\Delta\mathbf{x}(\tau)\|^2 d\tau \neq 0.$$

Then there exist an $\varepsilon > 0$, $t_m \geq T$ and a sequence $t_i \to \infty$ with $i = 1, 2, \cdots$ and $t_{i+1} \geq t_i + T$ such that $\int_{t_i-T}^{t_i} \|\Delta\mathbf{x}(\tau)\|^2 d\tau > \varepsilon$ when $t_i > t_m$. Hence from (7.18), we obtain

$$\lim_{t\to\infty} V(t, \Delta\mathbf{x}, \boldsymbol{\phi}) \leq V(T, \Delta\mathbf{x}(T), \boldsymbol{\phi}(T)) - \lim_{i\to\infty} \sum_{j=1}^{i} \int_{t_j-T}^{t_j} \|\Delta\mathbf{x}(\tau)\|^2 d\tau.$$

Since $V(T, \Delta\mathbf{x}(T), \boldsymbol{\phi}(T))$ is finite, the above relation implies $\lim_{t\to\infty} V(t, \Delta\mathbf{x}, \boldsymbol{\phi}) = -\infty$, a contradiction to the non-negativeness property of Lyapunov-Krasovskii functional $V(t, \Delta\mathbf{x}, \boldsymbol{\phi}) \geq 0$. □

7.4 Robustification and Extension

7.4.1 Learning With Projection

In many control applications, the upper and lower bounds of unknown system parameters are known *a priori*. In such circumstances, the parametric learning law (7.5) can be modified as

$$\begin{aligned} \hat{\boldsymbol{\theta}}(t) &= \mathcal{P}(\hat{\boldsymbol{\theta}}(t-T)) + k(t)\sigma(t)\boldsymbol{\xi}(t, \Delta\mathbf{x}), \\ \hat{\boldsymbol{\theta}}(t) &= \mathbf{0}, \forall t \in [-T,\ 0], \end{aligned} \tag{7.19}$$

where $\mathcal{P}(\hat{\boldsymbol{\theta}}) = [\mathcal{P}(\hat{\theta}_1), \cdots, \mathcal{P}(\hat{\theta}_i), \mathcal{P}(\hat{\theta}_m)]^T$ and the projection operator $\mathcal{P}(\hat{\theta}_i)$ is defined as

$$\mathcal{P}(\hat{\theta}_i) = \begin{cases} \hat{\theta}_i, & |\hat{\theta}_i| \leq \theta_i^* \\ p(\hat{\theta}_i), & |\hat{\theta}_i| > \theta_i^* \end{cases} \tag{7.20}$$

with θ_i^* the known upper bound for the parameter $\theta_i(t)$. $p(\hat{\theta}_i) \in \mathcal{C}^1(\mathcal{R}; \mathcal{R}^1)$ is a polynomial and satisfying $p(\theta_i^*) = \theta_i^*$, $p(-\hat{\theta}_i) = -p(\hat{\theta}_i)$, $0 \leq \frac{\partial p}{\partial \hat{\theta}_i} \leq 1$, $\frac{\partial p}{\partial \hat{\theta}_i}|_{\theta_i^*} = 1$ and the limit $\lim_{\hat{\theta}_i\to\infty} p(\hat{\theta}_i)$ is a constant. Figure 7.2 shows the shape of the projection operator.

By incorporating the additional system bounding information in the repetitive learning controller, our concern is whether the control performance could be improved. In the following we show that the control law (7.4) and the parametric learning law (7.19) with projection lead to the *uniform* convergence of the tracking error, instead of the *integral* convergence shown in Theorem 7.1.

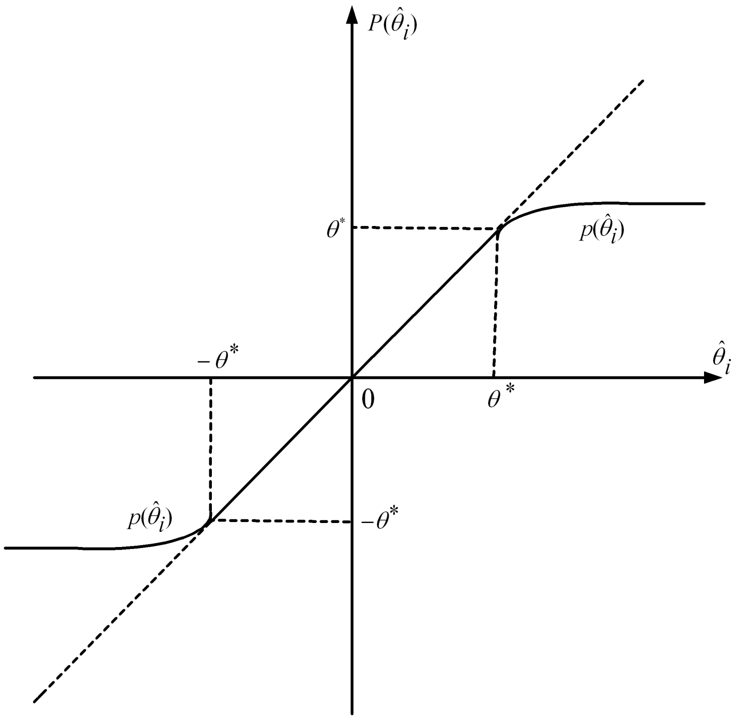

Figure 7.2. The definition of $\mathcal{P}(\hat{\theta})$.

Theorem 7.2. *For system (7.1) under Assumption 7.1, the control law (7.4) with the parametric learning law (7.19) guarantees the existence of solution and the uniformly asymptotical convergence of the tracking error* $\Delta\mathbf{x}$.

Proof. The solution $(\Delta\mathbf{x}, \hat{\boldsymbol{\theta}})$ of the dynamic system (7.9) for $t \in [0, T)$ is the same as the previous case in Theorem 7.1 without projection, because $\hat{\boldsymbol{\theta}}(t-T) = 0$. To prove the existence of solution in $[T, \infty)$, define the same Lyapunov-Krasovskii functional in (7.13). The relations (7.14) and (7.15) still hold as the projection operation is not directly involved. Next look at the relation (7.17), which might be affected by the introduction of the projection operator. We can easily verify the property

$$(\boldsymbol{\theta} - \hat{\boldsymbol{\theta}}_\ominus)^T(\boldsymbol{\theta} - \hat{\boldsymbol{\theta}}_\ominus) \geq (\boldsymbol{\theta} - \mathcal{P}(\hat{\boldsymbol{\theta}}_\ominus))^T(\boldsymbol{\theta} - \mathcal{P}(\hat{\boldsymbol{\theta}}_\ominus)),$$

for any quantities $\hat{\boldsymbol{\theta}}$. Using the parametric learning law (7.19), the periodic property

$\boldsymbol{\theta} = \boldsymbol{\theta}_\ominus$, the algebraic relation (7.16), and the above inequality, we have

$$\begin{aligned}\frac{1}{2q}(\boldsymbol{\phi}^T\boldsymbol{\phi} - \boldsymbol{\phi}_\ominus^T\boldsymbol{\phi}_\ominus) &\leq \frac{1}{2q}[(\boldsymbol{\theta} - \hat{\boldsymbol{\theta}})^T(\boldsymbol{\theta} - \hat{\boldsymbol{\theta}}) - (\boldsymbol{\theta} - \mathcal{P}(\hat{\boldsymbol{\theta}}_\ominus)^T(\boldsymbol{\theta} - \mathcal{P}(\hat{\boldsymbol{\theta}}_\ominus)] \\ &= \frac{1}{2q}[-2(\boldsymbol{\theta} - \hat{\boldsymbol{\theta}})^T(\hat{\boldsymbol{\theta}} - \mathcal{P}(\hat{\boldsymbol{\theta}}_\ominus)) - (\hat{\boldsymbol{\theta}} - \mathcal{P}(\hat{\boldsymbol{\theta}}_\ominus))^T(\hat{\boldsymbol{\theta}} - \mathcal{P}(\hat{\boldsymbol{\theta}}_\ominus))] \\ &= -\sigma\boldsymbol{\phi}^T\boldsymbol{\xi} - \frac{1}{2q}\sigma^2\boldsymbol{\xi}^T\boldsymbol{\xi}.\end{aligned}$$

which turns out to be the same as (7.17). In the sequel, the existence of solution and the integral convergence of $\Delta\mathbf{x}$ can be obtained according to Theorem 7.1.

According to the dynamic system (7.1), the control law (7.4), the parametric learning law (7.19) and in particular the projection, the boundedness of $\Delta\mathbf{x}$ ensures the finiteness of $\hat{\boldsymbol{\theta}}$, u and $\Delta\dot{\mathbf{x}}$. The boundedness of $\Delta\dot{\mathbf{x}}$ implies the uniform continuity of $\Delta\mathbf{x}$, thereafter the uniform continuity of the tracking error $\Delta\mathbf{x}$. As a result, $\lim_{t\to\infty}\|\Delta\mathbf{x}\| = 0$ uniformly. □

7.4.2 Learning With Damping

When the parameter bounds are not available, an alternative approach is the introduction of a damping (forgetting) factor. Note that the original parametric learning law (7.5) is a pointwise integrator, that is, for any $t \in [(i-1)T,\ iT)$, it performs discrete-time integration over the time sequence $t - iT$ for $i = 1, 2, \cdots, i-1$. Such an integral mechanism might be sensitive to many non-ideal factors, such as biased measurement noise, the unmodeled higher order dynamics, etc. A popular modification is to add a "damping" term such that the parametric updating mechanism becomes a low pass filter instead of an integrator. The parametric learning law (7.5) is modified as

$$\begin{aligned}\hat{\boldsymbol{\theta}}(t) &= \gamma\hat{\boldsymbol{\theta}}(t-T) + k(t)\sigma(t)\boldsymbol{\xi}(t, \Delta\mathbf{x}), \\ \hat{\boldsymbol{\theta}}(t) &= \mathbf{0}, \forall t \in [-T,\ 0],\end{aligned} \tag{7.21}$$

where $0 < \gamma < 1$ is the damping coefficient or the forgetting factor. In the following we derive the property of the closed-loop system under the new learning control law.

Theorem 7.3. *For system (7.1), under Assumption 7.1, the control law (7.4) with the parametric learning law (7.21) guarantees the finiteness of the solution trajectory* $(\Delta\mathbf{x}, \hat{\boldsymbol{\theta}})$ *in the large.*

Proof. The solution $\Delta\mathbf{x}$ for $t \in [0, T)$ is the same as the previous case in Theorem 7.1 without damping, because $\hat{\boldsymbol{\theta}}(t-T) = 0$. Thus in the following we discuss the solution in the interval $[T, \infty)$. Analogous to Theorem 7.1, assume the solution exists in $[T,\ (i-1)T)$ and can be continued up to $t_i \in [(i-1)T,\ iT)$. We need only to show the finiteness of the solution for any $t_i \in [(i-1)T,\ iT)$. Define the same Lyapunov-Krasovskii functional as (7.13) in Theorem 7.1. The relations (7.14) and (7.15) still hold as only the closed-loop dynamics is directly involved in the derivation. Next look at the relation (7.17), which is affected by the introduction of the damping factor. Using the parametric learning law (7.21), the periodic property $\boldsymbol{\theta} = \boldsymbol{\theta}_\ominus$ and the algebraic relation (7.16), we have

$$
\begin{aligned}
&\frac{1}{2q}(\boldsymbol{\phi}^T\boldsymbol{\phi} - \boldsymbol{\phi}_\ominus^T\boldsymbol{\phi}_\ominus) \\
&= \frac{1}{2q}[-2(\boldsymbol{\theta} - \hat{\boldsymbol{\theta}})^T(\hat{\boldsymbol{\theta}} - \hat{\boldsymbol{\theta}}_\ominus) - (\hat{\boldsymbol{\theta}} - \hat{\boldsymbol{\theta}}_\ominus)^T(\hat{\boldsymbol{\theta}} - \hat{\boldsymbol{\theta}}_\ominus)] \\
&= -\frac{1}{q}(\boldsymbol{\theta} - \hat{\boldsymbol{\theta}})^T(\hat{\boldsymbol{\theta}} - \gamma\hat{\boldsymbol{\theta}}_\ominus) + \frac{1}{q}(1-\gamma)(\boldsymbol{\theta} - \hat{\boldsymbol{\theta}})^T\hat{\boldsymbol{\theta}}_\ominus - \frac{1}{2q}(\hat{\boldsymbol{\theta}} - \hat{\boldsymbol{\theta}}_\ominus)^T(\hat{\boldsymbol{\theta}} - \hat{\boldsymbol{\theta}}_\ominus)
\end{aligned} \tag{7.22}
$$

The first term on the right hand side of (7.22), by substituting the parametric learning law (7.21), is $-\sigma\boldsymbol{\phi}^T\boldsymbol{\xi}$ which will cancel out the same term but with opposite sign in (7.15). In order to evaluate last two terms on the right hand side of (7.22), let us derive the following inequality. Define vectors $\mathbf{a}$, $\mathbf{b}$ and $\mathbf{c}$ with the same dimensions, then

$$\begin{aligned}(\mathbf{a}-\mathbf{b})^T\mathbf{c} &\leq \|\mathbf{a}\|\cdot\|\mathbf{c}\| - \mathbf{b}^T\mathbf{c} \\ &\leq \frac{1}{2}(\mathbf{a}^T\mathbf{a}+\mathbf{c}^T\mathbf{c}) - \mathbf{b}^T\mathbf{c} \\ &= \frac{1}{2}\mathbf{a}^T\mathbf{a}+\frac{1}{2}\mathbf{c}^T\mathbf{c}-\mathbf{b}^T\mathbf{c}+\frac{1}{2}\mathbf{b}^T\mathbf{b}-\frac{1}{2}\mathbf{b}^T\mathbf{b} \\ &= \frac{1}{2}\mathbf{a}^T\mathbf{a}+\frac{1}{2}(\mathbf{b}-\mathbf{c})^T(\mathbf{b}-\mathbf{c})-\frac{1}{2}\mathbf{b}^T\mathbf{b}. \end{aligned} \tag{7.23}$$

Using the above relationship, the last two terms on the right-side hand of (7.22) is

$$\begin{aligned}&\frac{1}{q}(1-\gamma)(\boldsymbol{\theta}-\hat{\boldsymbol{\theta}})^T\hat{\boldsymbol{\theta}}_\ominus - \frac{1}{2q}(\hat{\boldsymbol{\theta}}-\hat{\boldsymbol{\theta}}_\ominus)^T(\hat{\boldsymbol{\theta}}-\hat{\boldsymbol{\theta}}_\ominus) \\ &\leq \frac{1-\gamma}{2q}\boldsymbol{\theta}^T\boldsymbol{\theta}+\frac{1-\gamma}{2q}(\hat{\boldsymbol{\theta}}-\hat{\boldsymbol{\theta}}_\ominus)^T(\hat{\boldsymbol{\theta}}-\hat{\boldsymbol{\theta}}_\ominus)-\frac{1-\gamma}{2q}\hat{\boldsymbol{\theta}}^T\hat{\boldsymbol{\theta}}-\frac{1}{2q}(\hat{\boldsymbol{\theta}}-\hat{\boldsymbol{\theta}}_\ominus)^T(\hat{\boldsymbol{\theta}}-\hat{\boldsymbol{\theta}}_\ominus) \\ &\leq \frac{1-\gamma}{2q}(\boldsymbol{\theta}^T\boldsymbol{\theta}-\hat{\boldsymbol{\theta}}^T\hat{\boldsymbol{\theta}})\end{aligned}$$

Therefore, the upper right hand derivative of V is

$$\begin{aligned}\dot{V} &\leq -\frac{\lambda_Q}{2}\|\Delta\mathbf{x}\|^2+\frac{1-\gamma}{2q}(\boldsymbol{\theta}^T\boldsymbol{\theta}-\hat{\boldsymbol{\theta}}^T\hat{\boldsymbol{\theta}}) \\ &\leq -\frac{\lambda_Q}{2}\|\Delta\mathbf{x}\|^2-\frac{1-\gamma}{2q}\|\hat{\boldsymbol{\theta}}\|^2+\frac{1-\gamma}{2q}\|\boldsymbol{\theta}\|_s^2.\end{aligned} \tag{7.24}$$

Now we can show the finiteness of V in the interval $[(i-1)T,\ t_i)$. If V is finite at $(i-1)T$, then it remains finite at t_i because the maximum increasing rate of V is uniformly bounded by $\frac{1-\gamma}{2q}\|\boldsymbol{\theta}\|_s^2$. Consequently $\Delta\mathbf{x}$ remains finite. The finiteness of $\hat{\boldsymbol{\theta}}$ in the interval $[(i-1)T,\ t_i)$ can be derived from the finiteness of $\sigma(t)\boldsymbol{\xi}(t,\Delta\mathbf{x})$ in (7.21). This implies the solution $(\Delta\mathbf{x},\hat{\boldsymbol{\theta}})$ either remains bounded or tend to infinity as $t\to\infty$. Thus the solution $(\Delta\mathbf{x},\hat{\boldsymbol{\theta}})$ exists for $t\in[0,\ \infty)$.

We further show that the solution $(\Delta\mathbf{x},\hat{\boldsymbol{\theta}})$ cannot diverge to infinity as $t\to\infty$. From (7.24), $\dot{V}\leq 0$ as long as the solution $(\Delta\mathbf{x},\hat{\boldsymbol{\theta}})$ is outside a compact set $\mathcal{M}$ defined below

$$\mathcal{M}=\left\{(\Delta\mathbf{x},\hat{\boldsymbol{\theta}}):\frac{\lambda_Q}{2}\|\Delta\mathbf{x}\|^2+\frac{1-\gamma}{2q}\|\hat{\boldsymbol{\theta}}\|^2\geq\frac{1-\gamma}{2q}\|\boldsymbol{\theta}\|_s^2\right\}.$$

Define an ϵ-neighbourhood of $\mathcal{M}$ with $\epsilon>0$

$$\mathcal{M}_\epsilon=\left\{(\Delta\mathbf{x},\hat{\boldsymbol{\theta}}):\frac{\lambda_Q}{2}\|\Delta\mathbf{x}\|^2+\frac{1-\gamma}{2q}\|\hat{\boldsymbol{\theta}}\|^2\geq\frac{1-\gamma}{2q}\|\boldsymbol{\theta}\|_s^2+\epsilon\right\},$$

then $\dot{V} \leq -\epsilon$ for any $(\Delta\mathbf{x}, \hat{\boldsymbol{\theta}}) \in \mathcal{M}_\epsilon^c$ where $\mathcal{M}_\epsilon^c$ is the complementary set of $\mathcal{M}_\epsilon$. Since the solution exists in $[0,\ \infty)$, there is no finite escape time for $(\Delta\mathbf{x}, \hat{\boldsymbol{\theta}})$. First assume that $\Delta\mathbf{x}$, thereby V, diverges asymptotically. Consider the fact that $\dot{V} \leq \frac{1-\gamma}{2q}\|\boldsymbol{\theta}\|_s^2$, there must exist an infinite time interval $[t_s,\ \infty)$, such that

$$\frac{\lambda_Q}{2}\|\Delta\mathbf{x}\|^2 + \frac{1-\gamma}{2q}\|\hat{\boldsymbol{\theta}}\|^2 \in \mathcal{M}_\epsilon^c \qquad \forall t \in [t_s,\ \infty).$$

Since the solution exists in $[0,\ \infty)$, $V(t_s, \Delta\mathbf{x}(t_s), \boldsymbol{\phi}(t_s))$ is finite. Integrating $\dot{V}$ in (7.24) from $t \geq t_s$ we have

$$\lim_{t\to\infty} V(t) \leq V(t_s, \Delta\mathbf{x}(t_s), \boldsymbol{\phi}(t_s)) - \lim_{t\to\infty}\int_{t_s}^{t} \epsilon d\tau \to -\infty,$$

that is however impossible because $V \geq 0$. We can conclude that $\Delta\mathbf{x}$ cannot stay infinitely long in $\mathcal{M}_\epsilon^c$, and will always re-enter $\mathcal{M}_\epsilon$ after a finite interval. Hence $\Delta\mathbf{x}$ remains finite when $t \to \infty$. Note that the finiteness of $\Delta\mathbf{x}$ warrants the finiteness of $\sigma(t)\boldsymbol{\xi}(t, \Delta\mathbf{x})$ over the entire horizon $[0,\ \infty)$. On the other hand, the parametric learning law (7.21) with the damping γ is an asymptotically stable first order difference equation subject to the input $\sigma(t)\boldsymbol{\xi}(t, \Delta\mathbf{x})$. Therefore $\hat{\boldsymbol{\theta}}$ remains finite when $t \to \infty$. □

Remark 7.1. *Using an appropriate Lyapunov function, the adaptive control with the robust adaption law enhanced by a damping term achieves the asymptotical convergence to a compact set specified by the damping coefficient (Ioannou and Sun, 1996). Here in the repetitive learning control, we are dealing with rapidly time-varying parameters and a Lyapunov-Krasovskii functional is used. It would be difficult to derive such compact set with the functional as it does not warrant a uniform bound for the solution even if the functional itself is bounded. Nevertheless, if* γ *is chosen sufficiently close to* 1, *the integral convergence* $\lim_{t\to\infty}\int_{t-T}^{t}\|\Delta\mathbf{x}\|^2 d\tau = 0$ *can be achieved as we have shown in Theorem 7.1. Thus learning with damping provides more options, and one may decide the damping coefficient* γ *according to the control requirements.*

7.4.3 Extension to More General Cases

In this subsection, we extend the dynamic system (7.1) to a more general class described below

$$\begin{aligned} \dot{x}_j &= x_{j+1}, \qquad\qquad j = 1, 2, \cdots, n-1, \\ \dot{x}_n &= \boldsymbol{\theta}^T(t)\boldsymbol{\xi}(t, \mathbf{x}) + b(t, \mathbf{x})u(t), \quad \mathbf{x}(0) = \mathbf{x}_0. \end{aligned} \tag{7.25}$$

The presence of the input coefficient $b(t, \mathbf{x})$ makes the control task much more difficult to address. Note that if $b(t, \mathbf{x})$ is a known nonsingular function, the control problem is trivial because we can simply multiply the preceding repetitive learning control law by a factor $b^{-1}(t, \mathbf{x})$. In the following we focus on two cases with an unknown input coefficient.

<u>Case 1.</u> $b(t, \mathbf{x}) = b$ is an unknown constant but the sign is known *a priori*.

Without loss of generality, assume that $b > 0$. The error dynamics is

$$\begin{aligned} \Delta\dot{\mathbf{x}} &= A\Delta\mathbf{x} + \mathbf{b}[\mathbf{c}\Delta\mathbf{x} + \boldsymbol{\theta}^T\boldsymbol{\xi} + bu(t) - s(\mathbf{x}_r, r, t)] \\ &= A\Delta\mathbf{x} + b\mathbf{b}[b^{-1}\boldsymbol{\theta}^T\boldsymbol{\xi} + b^{-1}\mathbf{c}\Delta\mathbf{x} - b^{-1}s(\mathbf{x}_r, r, t) + u(t)]. \end{aligned} \tag{7.26}$$

Now define the extended parameter vector $\bar{\boldsymbol{\theta}}(t) = [b^{-1}\boldsymbol{\theta}(t)^T,\ b^{-1}]^T \in \mathcal{R}^{m+1}$, the extended regressor $\bar{\boldsymbol{\xi}}(t, \Delta\mathbf{x}) = [\boldsymbol{\xi}(t, \Delta\mathbf{x}),\ \mathbf{c}\Delta\mathbf{x} - s(\mathbf{x}_r, r, t)]^T \in \mathcal{R}^{m+1}$, and the new control law

$$\begin{aligned} u(t) &= -\hat{\bar{\boldsymbol{\theta}}}(t)^T\bar{\boldsymbol{\xi}}(t, \Delta\mathbf{x}) \\ \hat{\bar{\boldsymbol{\theta}}}(t) &= \hat{\bar{\boldsymbol{\theta}}}(t-T) + k(t)\sigma(t)\bar{\boldsymbol{\xi}}(t, \Delta\mathbf{x}), \\ \hat{\bar{\boldsymbol{\theta}}}(t) &= \mathbf{0}, \forall t \in [-T,\ 0], \end{aligned}$$

From (7.26), substituting the new control law and using the extended $\bar{\boldsymbol{\theta}}$ and $\bar{\boldsymbol{\xi}}$, the closed-loop error dynamics is

$$\begin{aligned} \Delta\dot{\mathbf{x}} &= A\Delta\mathbf{x} + b\mathbf{b}[\bar{\boldsymbol{\theta}}^T\bar{\boldsymbol{\xi}} - \hat{\bar{\boldsymbol{\theta}}}^T\bar{\boldsymbol{\xi}}] \\ &= A\Delta\mathbf{x} + b\mathbf{b}\bar{\boldsymbol{\phi}}\bar{\boldsymbol{\xi}} \end{aligned} \tag{7.27}$$

where $\bar{\boldsymbol{\phi}}(t) = \bar{\boldsymbol{\theta}}(t) - \hat{\bar{\boldsymbol{\theta}}}(t)$.

Define a new Lyapunov-Krasovskii functional

$$V(t, \Delta\mathbf{x}, \bar{\boldsymbol{\phi}}) = \frac{1}{2b}\Delta\mathbf{x}^T P\Delta\mathbf{x} + \frac{1}{2q}\int_{t-T}^{t} \bar{\boldsymbol{\phi}}^T(\tau)\bar{\boldsymbol{\phi}}(\tau)d\tau.$$

The upper right hand derivative of V is

$$\dot{V} = \frac{1}{2b}(\Delta\dot{\mathbf{x}}^T P\Delta\mathbf{x} + \Delta\mathbf{x}^T P\Delta\dot{\mathbf{x}}) + \frac{1}{2q}(\bar{\boldsymbol{\phi}}^T\bar{\boldsymbol{\phi}} - \bar{\boldsymbol{\phi}}_\ominus^T\bar{\boldsymbol{\phi}}_\ominus), \tag{7.28}$$

The first term on the right side of (7.28), in terms of (7.27), is

$$\frac{1}{2b}(\Delta\dot{\mathbf{x}}^T P\Delta\mathbf{x} + \Delta\mathbf{x}^T P\Delta\dot{\mathbf{x}}) \leq -\frac{\lambda_Q}{2b}\|\Delta\mathbf{x}(t)\|^2 + \sigma\bar{\boldsymbol{\phi}}^T\bar{\boldsymbol{\xi}}. \tag{7.29}$$

Clearly, (7.29) has the similar form as (7.15). Analogously, following the procedure in Theorem 7.1 we can further derive

$$\frac{1}{2q}(\bar{\boldsymbol{\phi}}^T\bar{\boldsymbol{\phi}} - \bar{\boldsymbol{\phi}}_\ominus^T\bar{\boldsymbol{\phi}}_\ominus) = -\sigma\bar{\boldsymbol{\phi}}^T\bar{\boldsymbol{\xi}} - \frac{q}{2}\sigma^2\bar{\boldsymbol{\xi}}^T\bar{\boldsymbol{\xi}}. \tag{7.30}$$

Substituting (7.29) and (7.30) into (7.28) yields

$$\dot{V} \leq -\frac{\lambda_Q}{2b}\|\Delta\mathbf{x}\|^2,$$

which is the same as (7.18) except for a constant $b > 0$. Therefore, the existence of solution and the convergence property can be derived exactly the same as in Theorem 7.1.

<u>Case 2.</u> $b(t, \mathbf{x}) = b(t) \in \mathcal{C}^1_{PT}([0,\infty); \mathcal{R}^1)$ is nonsingular with its sign known *a priori*.

Without loss of generality, assume $b(t) > 0$. Define a new quantity $\sigma = \mathbf{c}\Delta\mathbf{x}$, and vector $\mathbf{c}_1 = [0, c_1, \cdots, c_{n-1}]$. We can deal with the case by revising the control law (7.4) into

$$u(t) = -\beta\sigma - \hat{\bar{\boldsymbol{\theta}}}(t)^T\bar{\boldsymbol{\xi}}(t, \Delta\mathbf{x}),$$

where $\beta > 0$ is a feedback gain, $\hat{\bar{\boldsymbol{\theta}}}(t)$ is the estimate of the extended parametric vector $\bar{\boldsymbol{\theta}}(t) = [b^{-1}(t)\boldsymbol{\theta}(t),\ b^{-1}(t),\ b^{-2}(t)\dot{b}(t)]^T \in \mathcal{C}^1_{PT}([0,\infty); \mathcal{R}^{m+2})$, and the extended regressor is $\bar{\boldsymbol{\xi}}(t, \Delta\mathbf{x}) = [\boldsymbol{\xi}(t, \Delta\mathbf{x}),\ \mathbf{c}_1\Delta\mathbf{x} - s(\mathbf{x}_r, r, t),\ -\frac{1}{2}\sigma]^T \in \mathcal{R}^{m+2}$. The corresponding parametric learning law is

$$\begin{aligned}\hat{\bar{\boldsymbol{\theta}}}(t) &= \hat{\bar{\boldsymbol{\theta}}}(t-T) + k(t)\sigma\bar{\boldsymbol{\xi}}(t, \Delta\mathbf{x}),\\ \hat{\bar{\boldsymbol{\theta}}}(t) &= \mathbf{0}, \forall t \in [-T,\ 0].\end{aligned}$$

The Lyapunov-Krasovskii functional in this case is

$$V(t, \sigma, \bar{\boldsymbol{\phi}}) = \frac{1}{2}b^{-1}(t)\sigma^2 + \frac{1}{2q}\int_{t-T}^{t} \bar{\boldsymbol{\phi}}^T(\tau)\bar{\boldsymbol{\phi}}(\tau)d\tau$$

where $\bar{\boldsymbol{\phi}}(\tau) = \bar{\boldsymbol{\theta}}(\tau) - \hat{\bar{\boldsymbol{\theta}}}(\tau)$. The upper right hand derivative of the functional V is

$$\dot{V} = b^{-1}(t)\sigma\dot{\sigma} - \frac{1}{2}b^{-2}(t)\dot{b}(t)\sigma^2 + \frac{1}{2q}(\bar{\boldsymbol{\phi}}^T\bar{\boldsymbol{\phi}} - \bar{\boldsymbol{\phi}}_{\ominus}^T\bar{\boldsymbol{\phi}}_{\ominus}) \tag{7.31}$$

The first two terms on the right side of (7.31) can be rewritten as

$$\begin{aligned}&b^{-1}(t)\sigma\dot{\sigma} - \frac{1}{2}b^{-2}(t)\dot{b}(t)\sigma^2\\ &= b^{-1}(t)\sigma[\mathbf{c}_1\Delta\mathbf{x} + \boldsymbol{\theta}^T\boldsymbol{\xi} + b(t)u(t) - s(\mathbf{x}_r, r, t)] - \frac{1}{2}b^{-2}(t)\dot{b}(t)\sigma^2\\ &= -\beta\sigma^2 + \sigma[b^{-1}(t)\boldsymbol{\theta}^T\boldsymbol{\xi} + b^{-1}(t)(\mathbf{c}_1\Delta\mathbf{x} - s(\mathbf{x}_r, r, t)) + b^{-2}(t)\dot{b}(t)(-\frac{1}{2}\sigma) - \hat{\bar{\boldsymbol{\theta}}}^T\bar{\boldsymbol{\xi}}]\\ &= -\beta\sigma^2 + \sigma(\bar{\boldsymbol{\theta}} - \hat{\bar{\boldsymbol{\theta}}})^T\bar{\boldsymbol{\xi}} = -\sigma^2 + \sigma\bar{\boldsymbol{\psi}}^T\bar{\boldsymbol{\xi}}\end{aligned}$$

where $\bar{\boldsymbol{\xi}} = \bar{\boldsymbol{\xi}}(t, \Delta\mathbf{x})$. Analogously, following the procedure in Theorem 7.1 we can further derive

$$\frac{1}{2q}(\bar{\boldsymbol{\phi}}^T\bar{\boldsymbol{\phi}} - \bar{\boldsymbol{\phi}}_{\ominus}^T\bar{\boldsymbol{\phi}}_{\ominus}) = -\sigma\bar{\boldsymbol{\phi}}^T\bar{\boldsymbol{\xi}} - \frac{q}{2}\sigma^2\bar{\boldsymbol{\xi}}^T\bar{\boldsymbol{\xi}}.$$

Therefore

$$\dot{V} \leq -\beta\sigma^2,$$

from which we can derive the boundedness of σ and the integral convergence property

$$\lim_{t\to\infty}\int_{t-T}^{t}\sigma^2(\tau)d\tau = 0.$$

Notice that $\sigma = \mathbf{c}\Delta\mathbf{x}$ can be expressed as $\sigma = (D^{n-1}+c_{n-1}D^{n-2}+\cdots+c_2D+c_1)\Delta x_1$, where $(D^{n-1} + c_{n-1}D^{n-2} + \cdots + C_2D + c_1)$ is a stable polynomial of the differential operator $D \triangleq \frac{d}{dt}$. Therefore the boundedness of σ implies the boundedness of $\Delta\mathbf{x}$, therein the existence of solution $(\Delta\mathbf{x}, \hat{\tilde{\boldsymbol{\theta}}})$ in the large.

Note that the result of Case 2 can be extended to the input coefficient $b(t)b_1(t,\mathbf{x})$ with $b(t)$ defined same as Case 2 and $b_1(t,\mathbf{x})$ a known nonsingular function.

7.5 Illustrative Examples

Choose $\mathbf{c} = [1,\ 1]$, then $A = \begin{bmatrix} 0 & 1 \\ -1 & -1 \end{bmatrix}$. Choosing $Q = I_{2\times 2}$ to be an identity matrix, the solution of the Lyapunov equation is $P = \begin{bmatrix} 1.5 & 0.5 \\ 0.5 & 1 \end{bmatrix}$. Choose $k_1(t) = q(-\frac{2}{T^3}t^3 + \frac{3}{T^2}t^2)$, which is smooth and monotone between 0 and $q = 4$.

<u>*Case 1*</u>: Consider the system (7.1) where $\xi(t,\mathbf{x}) = x_1^2 x_2$ and parameter $\theta(t) = 1 + sin\pi t$ which has a periodicity $T = 2$. The given reference model is

$$\begin{aligned} \dot{x}_{r,1} &= x_{r,2}, \\ \dot{x}_{r,2} &= -1.1x_{r,1} - 0.4x_{r,2} - x_{r,1}^3 + 1.8\cos(1.8t). \end{aligned}$$

which is in fact a Duffing system producing a chaotic trajectory (non-periodic). The initial values are $\mathbf{x}(0) = [1,\ 0]^T$ and $\mathbf{x}_r(0) = [0,\ 1]^T$.

Applying the learning control (7.4) and the parametric learning law (7.5), the simulation results are shown in Figure 7.3 and Figure 7.4 respectively. In Figure 7.3, the horizontal axis denotes the number of periods and the vertical axis denotes $|\Delta x_i|_s$ over one period. The learning convergence can be clearly seen.

<u>*Case 2*</u>: In this case, there exists an unknown input coefficient $b(t) = 1 + cos^2(\pi t)$ which has the same periodicity $T = 2$. Applying the corresponding repetitive learning control law presented in Case 2, Part C of Section 7.3, simulation results

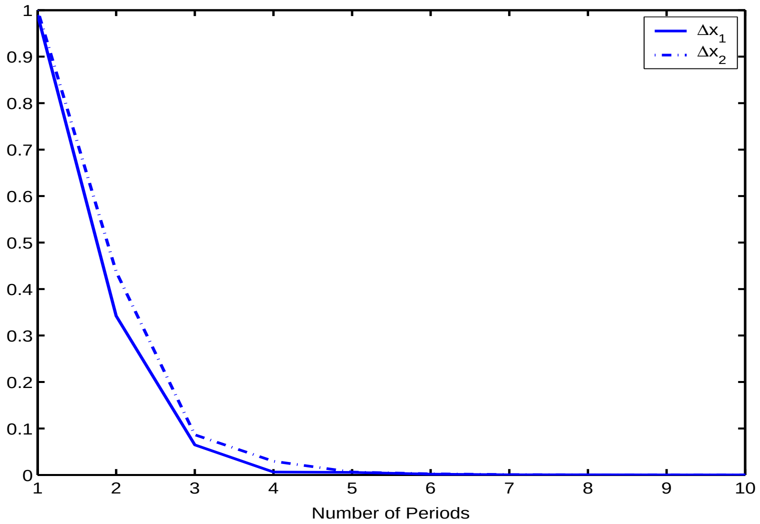

Figure 7.3. Learning convergence of the tracking errors (Case 1)

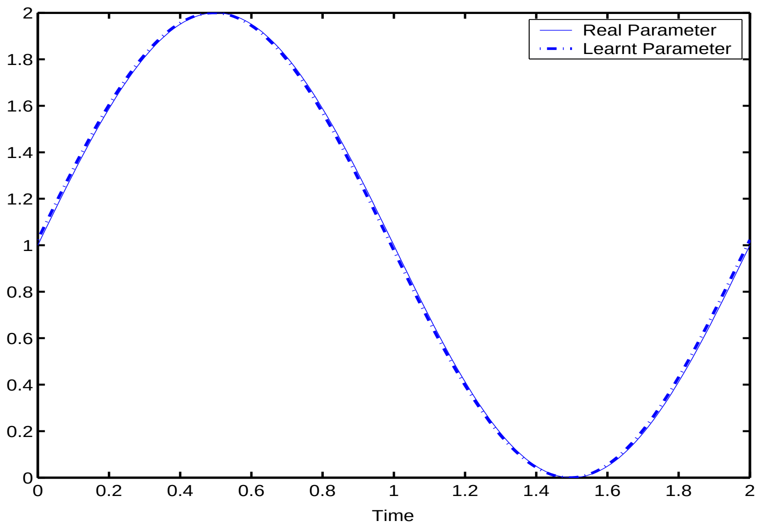

Figure 7.4. True and learnt parameters at 10−th period (Case 1)

are shown in Figure 7.5 and Figure 7.6 respectively. Note that the unknown parameters are $\bar{\boldsymbol{\theta}}(t) = [b^{-1}(t)\theta(t), b^{-1}(t), b^{-2}(t)\dot{b}(t)]$. Figure 7.6 only displays the parameter learning for the parameter $b^{-1}(t)\theta(t)$. From the figures, the tracking error convergence can be clearly seen. On the other hand, parameter learning convergence cannot be guaranteed in general.

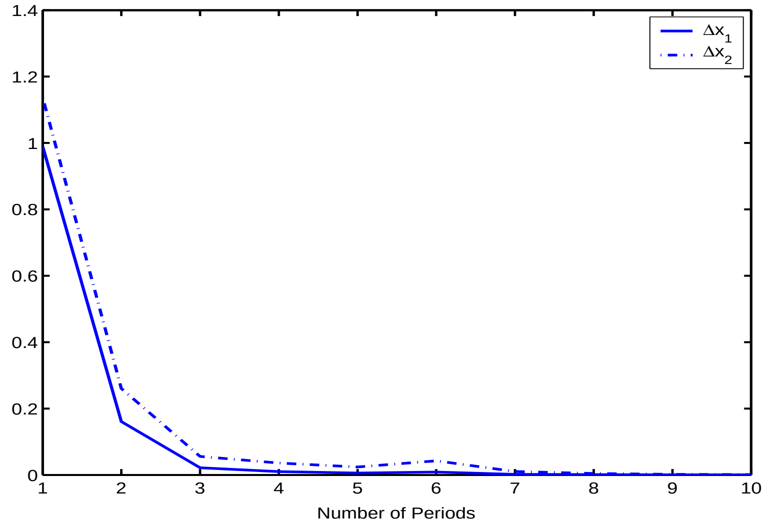

Figure 7.5. Learning convergence of the tracking errors (Case 2)

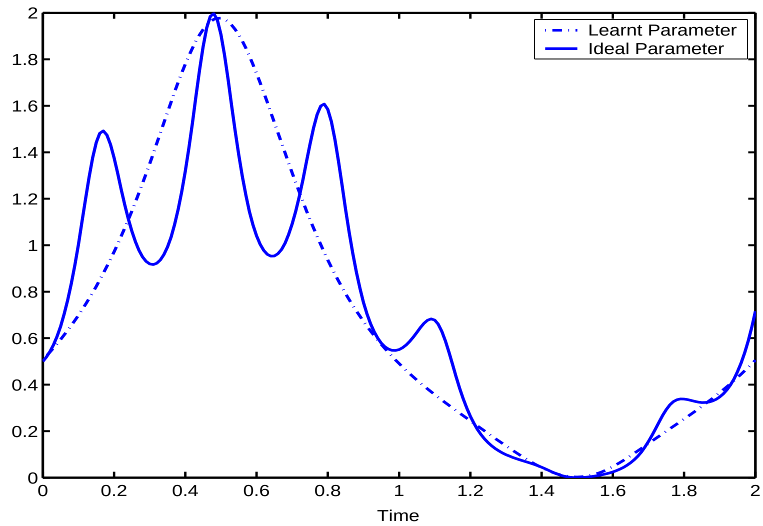

Figure 7.6. True and learnt parameters at 10th period (Case 2)

7.6 conclusion

In this chapter, new nonlinear learning control methods are developed for systems with unknown periodic parameters. With mathematical rigorousness the existence of solution and learning convergence are proved. Robustifying the nonlinear learning control with projection and forgetting factor has also been exploited in a systematic manner via the Lyapunov-Krasovskii functional approach.

Chapter 8

Repetitive Learning Control for Nonlinear Systems with Non-parametric Uncertainties

8.1 Introduction

Learning control aims at achieving the desired system performance via directly updating the control input, either repeatedly over a fixed finite time interval, or repetitively (cyclically) over an infinite time interval.

The concept of repetitive control was first proposed in (Hara *et al.*, 1988) for LTI systems and the convergence analysis was conducted in frequency domain using small gain theorem. In (Rogers and Owens, 1992) and (Owens *et al.*, 1999), the stability analysis was conducted in the form of differential-difference equations for linear repetitive processes. In (Longman, 2000), some design issues were exploited for linear repetitive control. In (Messner and Bodson, 1995), an adaptive feed-forward control using internal model equivalence was developed, which deals with

LTI systems with an exogenous disturbance consisting of a finite number of sinusoidal functions, and the adaptation mechanism estimates the constant unknown coefficients.

The extension of repetitive control to nonlinear dynamics has also been exploited. In (Messner *et al.*, 1991), the learning control has been applied to identify and compensate for a nonlinear disturbance function which is represented as an integral of a predefined kernel function multiplied by an unknown influence function that is state independent. In (Vecchio *et al.*, 2003), a kind of adaptive learning control scheme was proposed for a class of feedback linearizable systems to track a periodic reference, and the problem can be converted into the learning of a finite number of Fourier coefficients. In (Dixon *et al.*, 2003), the repetitive learning control is applied to a class of nonlinear systems with matched periodic disturbance. Since the periodic disturbance is a time function, it can also be treated as an unknown periodic coefficient under the framework of adaptive control (Xu, 2004). Note that, above mentioned learning control schemes require the plant to be parameterizable and what is aimed is asymptotic convergence along the time horizon, hence they may also be regarded as some kinds of nonlinear adaptive control under the generalized framework of adaptive control theory. In (Cao and Xu, 2001), a repetitive learning control scheme was developed for nonlinear dynamics without parameterization. Nonlinear robust control is used together with the repetitive learning mechanism, hence it requires the upper bound knowledge of the lumped uncertaities.

Under the present theoretical framework of repetitive control, it would be difficult to deal with plants with unknown nonlinear components that are not parameterizable. It is necessary to seek a new learning control strategy, which is able to use the simple but effective delay-based mechanism to carry out the repetitive learning, meanwhile is able to deal with lumped nonlinear unknowns. Henceforth, our first objective in this chapter is to establish a new control strategy – repetitive learning control (RLC) for nonlinear systems with non-parametric uncertainties. The learnability of the traditional repetitive control, acquired via the delay-loop, can be retained by incorporating such a delay-loop into a nonlinear learning mechanism. Meanwhile, a nonlinear feedback law will have to be developed to stabilize the nonlinear dynamics.

The delay-based learning mechanism of RLC actually forms a continuous-time difference equation, and is of infinite dimensions. Considering the plant described by nonlinear differential equations, the repetitive learning control system is described by a set of mixed nonlinear differential and continuous-time difference equations. The Lyapunov function based methods, which are proven to be powerful for nonlinear ordinary differential equations and difference equations, cannot be applied. In fact, very few results were reported for this class of systems when the closed-loop stability, convergence and boudnedness are concerned, except for some local analysis result (Pepe and Verriest, 2003). When the existence of solution is concerned, the well established results hitherto were given by (Cruz and Hale, 1970) and (Hale and Pedro, 1977), which however focus on the continuous-time difference equations satisfying a contractive mapping.

Our second objective of this chapter, then, is to provide a rigorous and global analysis with regards to the existence of solution and learning convergence for the RLC system. The Lyapunov-Krasovskii functional is employed to show the boundedness of states for any finite learning cycles. By means of the mathematical induction method the result for finite cycles can be extended to the entire time horizon. Next, using the system smoothness property the problem is converted into a set of neutral functional differential equations and the existence of solution can be concluded. As a consequence of the above analysis we can further derive the learning convergence property.

When extending the RLC to more general systems in the triangular form without strict matching condition, we encounter specific difficulty: backstepping design is not applicable. The problem arises due to the continuous-time difference learning law which cannot be replaced by a differential equation. An obvious contrast is the adaptive control, in which both the plant and adaptation law are described by differential equations. In backstepping design, the differentiability of the control law is indispensable for continuous-time systems. To overcome this problem, the repetitive learning is integrated with robust adaptive control. Repetitive learning will be used in the final step when all subsystems are aggregated, and robust adaptive control will be used for first $n-1$ subsystems.

This chapter is organized as follows. In Section 8.2, the repetitive learning control problem is formulated first. In Section 8.3 the existence of solution and learning convergence properties are analyzed. In Section 8.4, two robustification schemes are discussed. In Section 8.5, RLC is extended to more general classes of plants including the unmatched. Two illustrative examples are given in Section 8.6, and the conclusion is given in Section 8.7.

8.2 Problem Formulation

Consider the following system

$$\begin{aligned} \dot{x}_j &= x_{j+1}, & j &= 1, 2, \cdots, n-1, \\ \dot{x}_n &= \eta(t, \mathbf{x}) + u(t), & \mathbf{x}(0) &= \mathbf{x}_0, \end{aligned} \tag{8.1}$$

where $\mathbf{x} = [x_1, x_2, \cdots, x_n]^T$, and $\eta(t, \mathbf{x})$ is a continuously differentiable function w.r.t. the arguments $\mathbf{x}$ and t. In particular $\eta(t, \mathbf{x})$ is a lumped, non-parameterizable, and Local Lipschitzian nonlinear function, for example, $\eta(t, \mathbf{x}) = x^2 \cos x$ or $\eta(t, \mathbf{x}) = \frac{x_2}{2 + \sin t + x_1^2}$.

The control objective is to track the target trajectory $\mathbf{x}_r(t)$ generated by

$$\begin{aligned} \dot{x}_{r,j} &= x_{r,j+1}, \qquad\qquad j = 1, 2, \cdots, n-1, \\ \dot{x}_{r,n} &= s(t, \mathbf{x}_r, r), \qquad \mathbf{x}_r(0) \end{aligned} \tag{8.2}$$

where $\mathbf{x}_r = [x_{r,1}, x_{r,2}, \cdots, x_{r,n}]^T$, $s(\mathbf{x}_r, r, t)$ is a known smooth function w.r.t. all arguments, r is a constant reference input, and $\mathbf{x}_r(0)$ is a vector of the initial states. The ideal control input, $u_r(t)$, can be computed directly from the relation

$$\dot{x}_{r,n}(t) = \eta(t, \mathbf{x}_r) + u_r(t) \tag{8.3}$$

with the initial values $\mathbf{x}_r(0)$. From (8.2), $\dot{x}_{r,n} = s(t, \mathbf{x}_r(t), r)$. Therefore the ideal control is $u_r(t) = s(\mathbf{x}_r(t), t, r) - \eta(t, \mathbf{x}_r)$, which is however not available because of the presence of the unknown $\eta(t, \mathbf{x}_r)$. The central task now is to learn the ideal control $u_r(t)$. As such, the learning objective shall be the quantity $u_r(t)$, that is, to learn the ideal control profile directly. As being known, the repetitive learning control is especially effective in dealing with periodic quantities. Thus if $u_r(t)$ is periodic, we may apply the repetitive learning control approach to solve the tracking problem.

Assumption 8.1. *The desired trajectory $\mathbf{x}_r(t)$, and the quantity $\eta(t, \mathbf{x}_r)$, are periodic with a periodicity T, namely, $\mathbf{x}_r(t) \in \mathcal{C}^2_{PT}([0, \infty); \mathcal{R}^n)$ and $\eta(t, \mathbf{x}_r) = \eta(t - T, \mathbf{x}_r)$.*

Remark 8.1. *Any homogeneous function $\eta(\mathbf{x})$ satisfies Assumption 8.1.*

From the periodicity of $\mathbf{x}_r(t)$, we can derive that $\dot{\mathbf{x}}_r \in \mathcal{C}^1_{PT}([0, \infty); \mathcal{R}^n)$ and $s(t, \mathbf{x}_r(t), r) \in \mathcal{C}^1_{PT}([0, \infty); \mathcal{R}^1)$. From the periodicity of $\mathbf{x}_r(t)$ and Assumption 8.1, $\eta(t, \mathbf{x}_r) \in \mathcal{C}^1_{PT}([0, \infty); \mathcal{R}^1)$. In the sequel, the ideal control $u_r(t) = s(t, \mathbf{x}_r(t), r) - \eta(t, \mathbf{x}_r)$, is a function in the space $\mathcal{C}^1_{PT}([0, \infty); \mathcal{R}^1)$. The principal idea of repetitive learning control method, therefore, shall be applicable for this class of periodic learning tasks.

However, a learning mechanism alone, characterized by the continuous-time difference equation, is difficult to solve the problem. We may note the discrepancy in initial conditions $\mathbf{x}(0) \neq \mathbf{x}_r(0)$. Even if $u_r(t)$ is directly achievable such that $u(t) = u_r(t)$ for $t \geq 0$, the nonlinear system (8.1) may not produce the desired response $\mathbf{x}_r$, what is more, it may even go divergence in a finite time. From the theory of differential equation, a nonlinear ODE may produce totally different solution trajectories under different initial conditions. We need a robust control mechanism working concurrently with the learning mechanism to guarantee the asymptotic stability of the closed-loop system.

In designing a robust feedback controller for the nonlinear system (8.1), the most popular approach is first to assume a upper bounding function $\alpha(t, \mathbf{x})$ for $\eta(t, \mathbf{x})$, e.g. $\alpha(t, \mathbf{x}) \geq |\eta(t, \mathbf{x})|$, then construct a feedback control law using the bounding function $\alpha(t, \mathbf{x})$. The min-max control (Corless and Leitmann, 1981) and sliding mode control (Yu and Xu, 2000) are representative approaches of robust feedback control. The bounding function $\alpha(t, \mathbf{x})$ shall be known *a priori* and can be highly nonlinear such as local Lipschitzian. Repetitive learning can be incorporated into the robust control loop (Cao and Xu, 2001). However, it should be noted that the robust control alone can work well in this circumstance, and the learning mechanism is an add-on to the existing robust control aiming at further improving the performance. In this chapter, we explore a new scenario in which the robust control alone is unable to ensure a stable closed-loop, thus the repetitive learning mechanism and the robust control mechanism have to be integrated, working jointly to warrant a stable control loop and meanwhile achieve learning convergence repetitively.

The new scenario is characterized by the following bounding condition.

Assumption 8.2.

$$|\eta(t,\mathbf{x}) - \eta(t,\mathbf{y})| \leq \alpha(t,\mathbf{x},\mathbf{y})\|\mathbf{x}-\mathbf{y}\|,$$

where $\alpha(t,\mathbf{x},\mathbf{y})$ *is a known bounding function.*

Assumption 8.2 implies that the "variation" of the local Lipschitzian function η with respect to $\mathbf{x}$ should be limited from above by a known bound which can also be any nonlinear function, e.g. local Lipschitzian function, of $\mathbf{x}$. Hence it is not a very strict constraint. Clearly, most existing robust control methods may not be suitable in this circumstance because a bound for the variation of η does not warrant a finite bound for η itself.

Let us construct the integrated controller. First formulate the error dynamics of $\Delta\mathbf{x} = \mathbf{x} - \mathbf{x}_r$. Define $\mathbf{b} = [0\ 0\ \cdots\ 0\ 1]^T$, and $\mathbf{c} = [c_1, c_2, \cdots, c_{n-1}, 1]$ is chosen such that

$$A \triangleq \begin{bmatrix} 0 & 1 & 0 & \cdots & 0 & 0 \\ 0 & 0 & 1 & \cdots & 0 & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots & \\ 0 & 0 & 0 & \cdots & 0 & 1 \\ -c_1 & -c_2 & -c_3 & \cdots & -c_{n-1} & -1 \end{bmatrix} \tag{8.4}$$

is an asymptotically stable matrix. Based on Lyapunov stability theory for LTI systems, for a given positive definite matrix $Q \in R^{n\times n}$, there exits a unique positive definite matrix $P \in R^{n\times n}$ satisfying the following Lyapunov equation

$$A^T P + PA = -Q.$$

Let λ_Q be the minimum eigenvalue of the matrix Q, $-\mathbf{w}^T Q\mathbf{w} \leq -\lambda_Q\|\mathbf{w}\|^2$ holds for any $\mathbf{w} \in R^n$.

From (8.1) and (8.3), the dynamics of $\Delta\mathbf{x}$ can be expressed as

$$\Delta\dot{\mathbf{x}} = A\Delta\mathbf{x} + \mathbf{b}(\mathbf{c}\Delta\mathbf{x} + \eta - \eta_r + u - u_r), \tag{8.5}$$

where $\eta_r = \eta(t, \mathbf{x}_r)$. The integrated repetitive learning control law is

$$\begin{aligned} u(t) &= \hat{u}(t) - \mathbf{c}\Delta\mathbf{x} - \frac{1}{\lambda_Q}\alpha^2(t, \mathbf{x}, \mathbf{x}_r)\sigma(t), & (8.6) \\ \hat{u}(t) &= \hat{u}(t-T) - k(t)\sigma(t), & (8.7) \\ \hat{u}(t) &= 0, \forall t \in [-T,\ 0], \end{aligned}$$

where $\sigma(t) = \mathbf{b}^T P \Delta\mathbf{x}$. $k(t)$ is the learning gain defined as

$$k(t) = \begin{cases} 0, & -T \leq t < 0, \\ k_1(t), & 0 \leq t < T, \\ k_0, & t \geq T, \end{cases} \tag{8.8}$$

where $k_0 > 0$ is a constant, $k_1(t)$ is chosen to be monotone and smooth such that $k(t)$ is a smooth function on $[-T, \infty)$.

Note that now the objective of repetitive learning is to directly learn the ideal control, that is, tune $\hat{u}(t)$ in (8.7) to approach $u_r(t)$. $-\frac{1}{\lambda_Q}\alpha^2(t, \mathbf{x}, \mathbf{x}_r)\sigma(t)$ in (8.6) constitutes the robust feedback.

8.3 Existence of Solution and Convergence

Denote $\alpha \triangleq \alpha(t, \mathbf{x}, \mathbf{x}_r)$ and $\nu \triangleq u_r - \hat{u}$. Substituting the learning control law (8.6) into the dynamics (8.5), the closed-loop error dynamics is

$$\Delta\dot{\mathbf{x}} = A\Delta\mathbf{x} + \mathbf{b}(\eta - \eta_r - \nu - \frac{1}{\lambda_Q}\alpha^2\sigma). \tag{8.9}$$

In the closed-loop dynamics, there are two unknown terms u_r and $\eta - \eta_r$. The first term will be compensated by $\hat{u}$ through repetitive learning. The second term $\eta - \eta_r$ will be compensated jointly by $A\Delta\mathbf{x}$ and the robust control $-\frac{1}{\lambda_Q}\alpha^2\sigma$.

From the error dynamics (8.9) and the updating law (8.6), we have

$$\begin{cases} \Delta\dot{\mathbf{x}} &= \mathbf{f}(t, \Delta\mathbf{x}, \hat{u}) \\ \hat{u}(t) &= \hat{u}(t-T) - k(t)\mathbf{b}^T P \Delta\mathbf{x}, \end{cases} \tag{8.10}$$

where

$$\mathbf{f}(t, \Delta\mathbf{x}, \hat{u}) \;\; = \;\; A\Delta\mathbf{x} + \mathbf{b}(\eta - \eta_r + \hat{u} - u_r - \frac{1}{\lambda_Q}\alpha^2\sigma).$$

The learning control system consists of neutral differential and continuous-time difference equations.

Theorem 8.1. *For the system (8.10) under Assumption 8.1 and Assumption 8.2, the learning control law (8.6) and (8.7) guarantees the existence of solution* $(\Delta\mathbf{x}, \hat{u})$ *in* $[0, \infty)$ *and asymptotical convergence*

$$\lim_{t\to\infty}\int_{t-T}^{t}\|\Delta\mathbf{x}\|^2 d\tau = 0.$$

Proof. Define the regions $\Omega_i \triangleq [(i-1)T,\ iT)\times R^n$ for $(t, \mathbf{x})$. The proof is composed of three parts. *Part 1* and *Part 2* prove the existence of solution $(\Delta\mathbf{x}, \hat{u})$ in the domain $[0,\ T)$ and $[T,\ \infty)$ respectively. *Part 3* derives the convergence property of the tracking error $\Delta\mathbf{x}$.

Part 1. Existence of the solution $(\Delta\mathbf{x}, \hat{u})$ *in* $[0,\ T)$

For $i = 1$, we have $\hat{u}(t) = \mathbf{0}$ for $t \in [-T,\ 0]$. Therefore, by substituting $\hat{u}(t)$ into $\mathbf{f}$ the dynamics (8.10) renders to a set of ODE (Ordinary Differential Equation), and $\mathbf{f}(t, \Delta\mathbf{x}, \hat{u}) : \Omega_1 \to R^n$ is continuous in $\Delta\mathbf{x}$ by virtue of the smoothness of η. By Peano's Existence Theorem (Zheng *et al.*, 1991), associated with the initial condition $\Delta\mathbf{x}(0)$, the equation (8.10) has a continuous solution in a neighborhood of $t = 0$. Furthermore it is easy to check that $\mathbf{f}(t, \Delta\mathbf{x}, \hat{u})$ is locally Lipschitzian in $\Delta\mathbf{x}$. We need only to consider the solution for $t > 0$. Assume $[0, t_1)$ be the maximal interval to which the solution $\Delta\mathbf{x}$ can be continued up. Proposition 7.1 implies that $\Delta\mathbf{x}$ tends to the boundary $\partial\Omega_1$ of Ω_1 as $t \to t_1$. It further implies that $\lim_{t\to t_1}\|\Delta\mathbf{x}\| = \infty$ if $t_1 \leq T$, i.e., for any $C > 0$, there exists $\delta_1 > 0$ such that $\|\Delta\mathbf{x}\| \geq C$ for all $t \geq t_1 - \delta_1$. Since $\Delta\mathbf{x}$ exists for all $t \in [0, t_1 - \frac{\delta_1}{2}]$, define the following Lyapunov-Krasovskii functional:

$$V(t, \Delta\mathbf{x}, \nu) = \frac{1}{2}\Delta\mathbf{x}^T P \Delta\mathbf{x} + \frac{1}{2q}\int_{t-T}^{t} \nu^2(\tau) d\tau. \tag{8.11}$$

Now we prove the finiteness of $V(t, \Delta\mathbf{x}, \nu)$ for all $t \in [0, t_1 - \frac{\delta_1}{2}]$. From the existence theorem of differential equation (Yoshizawa, 1975) there exists a $T_1 > 0$ and $[0, T_1) \subset [0, t_1 - \frac{\delta_1}{2}]$, the boundedness of $V(t, \Delta\mathbf{x}, \nu)$ over $[0, T_1)$ can be guaranteed and we need only focus on the interval $[T_1, t_1 - \frac{\delta_1}{2}]$. For any $t \in [T_1, t_1 - \frac{\delta_1}{2}]$, the upper right hand derivative of V is

$$\dot{V} = \frac{1}{2}(\Delta\dot{\mathbf{x}}^T P \Delta\mathbf{x} + \Delta\mathbf{x}^T P \Delta\dot{\mathbf{x}}) + \frac{1}{2q}(\nu^2 - \nu_\ominus^2),$$

where $\nu_\ominus = u_{r,\ominus} - \hat{u}_\ominus$, $u_{r,\ominus} = u_r(t - T)$ and $\hat{u}_\ominus = \hat{u}(t - T)$. Substitution of the tracking error dynamics (8.9) yields

$$\begin{aligned}
&\frac{1}{2}(\Delta\dot{\mathbf{x}}^T P \Delta\mathbf{x} + \Delta\mathbf{x}^T P \Delta\dot{\mathbf{x}}) \\
&\quad = -\frac{1}{2}\Delta\mathbf{x}^T Q \Delta\mathbf{x} + \sigma(\eta - \eta_r - \nu - \frac{1}{\lambda_Q}\alpha^2\sigma) \\
&\quad \le -\frac{\lambda_Q}{2}\|\Delta\mathbf{x}\|^2 + |\sigma| \cdot \alpha\|\Delta\mathbf{x}\| - \frac{1}{\lambda_Q}\alpha^2\sigma^2 - \sigma\nu \\
&\quad = -\frac{\lambda_Q}{4}\|\Delta\mathbf{x}\|^2 - \sigma\nu - (\frac{\sqrt{\lambda_Q}}{2}\|\Delta\mathbf{x}\| - \frac{1}{\sqrt{\lambda_Q}}\alpha|\sigma|)^2.
\end{aligned} \tag{8.12}$$

Since $\hat{u}_\ominus = \hat{u}(t - T) = 0$ for all $t \in [0,\ T)$, $\hat{u}(t) = -k_1(t)\sigma(t)$. From the definition of $k(t)$, $k_1(t)$ is strictly increasing in $[0, T)$, thus $\frac{1}{k_1(t)} \ge \frac{1}{k_0}$ is ensured in the time interval $[T_1,\ T)$. We have

$$\begin{aligned}
\frac{1}{2k_0}(\nu^2 - \nu_\ominus^2) &= \frac{1}{2k_0}(u_r - \hat{u})^2 - \frac{1}{2k_0}(u_{r,\ominus} - \hat{u}_\ominus)^2 \\
&\le \frac{1}{2k_1(t)}(u_r - \hat{u})^2 - \frac{1}{2k_0}u_{r,\ominus}^2 \\
&\le \frac{u_r^2}{2k_1(t)} - \frac{1}{k_1(t)}\hat{u}(u_r - \hat{u}) - \frac{\hat{u}^2}{2k_1(t)} \\
&\le \frac{u_r^2}{2k_1(t)} + \sigma\nu.
\end{aligned}$$

Therefore from (8.12) and above we obtain

$$\begin{aligned}\dot{V} &\leq -\frac{\lambda_Q}{4}\|\Delta\mathbf{x}\|^2 - \sigma\nu - (\frac{\sqrt{\lambda_Q}}{2}\|\Delta\mathbf{x}\| - \frac{1}{\sqrt{\lambda_Q}}\alpha|\sigma|)^2 \\ &\quad + \frac{u_r^2}{2k_1(t)} + \sigma\nu \\ &\leq \frac{u_r^2}{2k_1(t)},\end{aligned} \tag{8.13}$$

i.e.,

$$V(t, \Delta\mathbf{x}, \nu) \leq V(T_1, \Delta\mathbf{x}(T_1), \nu(T_1)) + \frac{1}{2}\int_{T_1}^{t} \frac{u_r^2(\tau)}{k_1(\tau)} d\tau.$$

Since $u_r(t) \in \mathcal{C}^1_{PT}([0,\infty); \mathcal{R}^1)$, $\int_{T_1}^{t} \frac{u_r^2(\tau)}{k_1(\tau)} d\tau$ is bounded for $t \in [T_1, T)$. Thus V is bounded for all $t \in [0, t_1 - \frac{\delta_1}{2}]$. Let $N^2\lambda_P > 0$ be the bound of V on $[0, t_1 - \frac{\delta_1}{2}]$, where λ_P is the minimum eigenvalue of the positive definite matrix P. Then N does not depend on δ_1. By the definition of Lyapunov functional V, we can see that $\|\Delta\mathbf{x}\| \leq \sqrt{V/\lambda_P} = N$ for all $t \in [0, t_1 - \frac{\delta_1}{2}]$. Taking $C = 2N$ in advance, for the corresponding $\delta_1 > 0$ we have

$$C \leq \|\Delta\mathbf{x}(t_1 - \frac{\delta_1}{2})\| \leq N = \frac{C}{2},$$

a contradiction which implies $t_1 \geq T$. This assures the solution $\Delta\mathbf{x}$ of the dynamic system (8.10) exists in $[0, T]$. Further, considering the smoothness of the right hand side of equation (8.10), $\Delta\mathbf{x}(t)$ and $\hat{u}(t)$ are both continuously differentiable for any $t \in [0, T)$.

Part 2. Existence of the solution $(\Delta\mathbf{x}, \hat{u})$ in $[T, \infty)$

Assume that the solution $\Delta\mathbf{x}$ and $\hat{u}$ of the differential difference equation (8.10) exists in $[(j-1)T, jT)$ for $j = 2, \cdots, i-1$. This implies both $\mathbf{x}$ and $\hat{u}$ are continuously differentiable for all $t \in [0,\ (i-1)T)$. Assume that the solution of (8.10) can be continued up to a time $t \in [(i-1)T, iT)$, by differentiating $\hat{u}$ we obtain

$$\begin{aligned}\Delta\dot{\mathbf{x}} &= \mathbf{f}(t, \Delta\mathbf{x}, \hat{u}), \qquad t \in [(i-1)T, iT), \\ \dot{\hat{u}}(t) &= g(t, \Delta\mathbf{x}, \hat{u}(t), \dot{\hat{u}}(t-T)),\end{aligned} \tag{8.14}$$

where

$$g(t, \Delta\mathbf{x}, \hat{u}(t), \dot{\hat{u}}(t-T)) = \dot{\hat{u}}(t-T) - k_0 \mathbf{b}^T P \mathbf{f}(t, \Delta\mathbf{x}, \hat{u}).$$

Note that the function $\mathbf{f}(t, \Delta\mathbf{x}, \hat{u})$ and $g(t, \Delta\mathbf{x}, \hat{u}(t), \dot{\hat{u}}(t-T))$ are continuous with respect to the arguments, and the solution $(\Delta\mathbf{x}, \hat{u})$ are continuously differentiable on $[(i-2)T, (i-1)T)$. For $t > T$, $\hat{u}_\ominus$ cannot be ignored in the updating law, and (8.14) is now truly a mixture of differential and continuous-time difference equations of neural type. According to Proposition 7.2, the solution $(\Delta\mathbf{x}, \hat{u})$ of the equation (8.14) exists at the neighborhood of the point $(i-1)T$. Furthermore, $\mathbf{f}(t, \Delta\mathbf{x}, \hat{u}) : \Omega_i \to R^n$ is continuous and locally Lipschitzian in $\Delta\mathbf{x}$ and $\hat{u}$. Thus the solution $\Delta\mathbf{x}$ can be continued up to the boundary $\partial\Omega_i$ of Ω_i. Let $[(i-1)T, t_i)$ be the maximal interval to which the solution $\Delta\mathbf{x}$ can be continued up. If $t_i \le iT$, there exists a $\delta_i > 0$ such that $\|\Delta\mathbf{x}\| \ge C$ for all $t \ge t_i - \delta_i$. For $t \in [(i-1)T, t_i - \frac{\delta_i}{2})$, define the Lyapunov-Krasovskii functional

$$V(t, \Delta\mathbf{x}, \nu) = \frac{1}{2}\Delta\mathbf{x}^T P \Delta\mathbf{x} + \frac{1}{2k_0}\int_{t-T}^{t} \nu^2 d\tau. \tag{8.15}$$

Then the upper right hand derivative of V is

$$\dot{V} = \frac{1}{2}(\Delta\dot{\mathbf{x}}^T P \Delta\mathbf{x} + \Delta\mathbf{x}^T P \Delta\dot{\mathbf{x}}) + \frac{1}{2k_0}(\nu^2 - \nu_\ominus^2) \tag{8.16}$$

For the first term on the right side of (8.16), the result of (8.12) still holds. Let us compute the second term on the right hand side of (8.16). Using the learning updating law (8.7), the periodic property $u_r = u_{r,\ominus}$, and the algebraic relationship

$$(a-b)^2 - (a-c)^2 = -2(a-b)(b-c) - (b-c)^2, \tag{8.17}$$

we have

$$\begin{aligned}
&\frac{1}{2k_0}[(u_r - \hat{u})^2 - (u_{r,\ominus} - \hat{u}_\ominus)^2] \\
&= \frac{1}{2k_0}[-2(u_r - \hat{u})(\hat{u} - \hat{u}_\ominus) - (\hat{u} - \hat{u}_\ominus)^2] \\
&= \sigma\nu - \frac{k_0}{2}\sigma^2.
\end{aligned} \tag{8.18}$$

Substituting (8.12) and (8.18) into (8.16), the upper right hand derivative of V is

$$\begin{aligned}\dot{V} &\leq -\frac{\lambda_Q}{4}\|\Delta\mathbf{x}\|^2 - \frac{k_0}{2}\sigma^2 \\ &\quad -(\frac{\sqrt{\lambda_Q}}{2}\|\Delta\mathbf{x}\| - \frac{1}{\sqrt{\lambda_Q}}\alpha|\sigma|)^2 \end{aligned} \tag{8.19}$$

Clearly $V(t, \Delta\mathbf{x}, \nu)$ will be bounded for $t \in [(i-1)T,\ t_i - \frac{1}{2}\delta_i)$ as far as $V(\tau, \Delta\mathbf{x}(\tau), \nu(\tau))$ is bounded for $\tau \in [0,\ (i-1)T)$. Let $N^2\lambda_P$ be the bound of V on $[(i-1)T, t_i - \frac{\delta_i}{2})$, then N does not depend on δ_i. By the definition of Lyapunov-Krasovskii functional, we have $\|\Delta\mathbf{x}(t)\| \leq \sqrt{V/\lambda_P} = N$ for all $t \in [(i-1)T,\ t_i)$. Taking $C = 2N$ in advance, if the solution can only be continued up to $t_i < iT$, then we again has the contradiction

$$C \leq \|\Delta\mathbf{x}(t_i - \frac{\delta_i}{2})\| \leq N = \frac{C}{2}.$$

According to the theory of mathematical induction, the solution $\Delta\mathbf{x}$ exists in $t \in [(i-1)T,\ iT)$ for any finite i. Furthermore, since the solution $\hat{u}(t)$ exists for $t \in [0,\ (i-1)T)$, then from

$$\hat{u}(t) = \hat{u}(t-T) + k(t)\mathbf{b}^T P\Delta\mathbf{x}$$

and the existence of $\Delta\mathbf{x}$ for $t \in [(i-1)T, iT)$, the solution $\hat{u}$ exists for $t \in [(i-1)T,\ iT)$. Thus the solution $\Delta\mathbf{x}$ and $\hat{u}$ exists in $[0,\ iT)$ for any finite i. This implies that the solution $(\Delta\mathbf{x},\ \hat{u})$ either is uniformly bounded or tends to infinity as $t \to \infty$. Thus $\Delta\mathbf{x}$ and $\hat{u}$ exist for $t \in [0, \infty)$.

Part 3. Asymptotical convergence

Now derive the integral convergence

$$\lim_{t\to\infty}\int_{t-T}^{t}\|\Delta\mathbf{x}\|^2 d\tau = 0$$

using the relation (8.19), that is, $\dot{V}$ is negative semi-definite for $t \in [T, \infty)$. Suppose that

$$\lim_{t\to\infty}\int_{t-T}^{t}\|\Delta\mathbf{x}\|^2 d\tau \neq 0.$$

Then there exist an $\varepsilon > 0$, $t_m \geq T$ and a sequence $t_i \to \infty$ with $i = 1, 2, \cdots$ and $t_{i+1} \geq t_i + T$ such that $\int_{t_i - T}^{t_i} \|\Delta \mathbf{x}\|^2 d\tau > \varepsilon$ when $t_i > t_m$. Hence from (8.19), we obtain

$$\begin{aligned}\lim_{t\to\infty} V(t, \Delta\mathbf{x}, \nu) \quad &\leq \quad V(T, \Delta\mathbf{x}(T), \nu(T)) \\ &\quad - \lim_{i\to\infty} \sum_{j=1}^{i} \int_{t_j - T}^{t_j} \|\Delta\mathbf{x}(\tau)\|^2 d\tau\end{aligned}$$

Since $V(T, \Delta\mathbf{x}(T), \nu(T))$ is finite, the above relation implies $\lim_{t\to\infty} V(t, \Delta\mathbf{x}, \nu) = -\infty$, a contradiction to the non-negativeness property of Lyapunov-Krasovskii functional $V(t, \Delta\mathbf{x}, \nu) \geq 0$. □

8.4 Robustification

8.4.1 Learning Control With Projection

From the point of view of practical implementation, $u_r(t)$ must be finite. If there exists a known constant u^* such that for the given $\mathbf{x}_r(t)$, $\max_t |u_r(t)| \leq u^*$, the updating law (8.7) can be modified as

$$\begin{aligned}\hat{u}(t) &= \mathcal{P}(\hat{u}(t-T)) - k(t)\sigma(t), \\ \hat{u}(t) &= 0, \quad \forall t \in [-T,\ 0],\end{aligned} \tag{8.20}$$

where the projection operator $\mathcal{P}(\hat{u})$ is defined as

$$\mathcal{P}(\hat{u}) = \begin{cases} \hat{u}, & |\hat{u}| \leq u^* \\ p(\hat{u}), & |\hat{u}| > u^*, \end{cases}$$

where $p(\hat{u}) \in \mathcal{C}^1(\mathcal{R}; \mathcal{R}^1)$ is a polynomial and satisfying $p(\hat{u}) = \hat{u}$, $p(-\hat{u}) = -p(\hat{u})$, $0 \leq \frac{\partial p}{\partial \hat{u}} \leq 1$, $\frac{\partial p}{\partial \hat{u}}|_{u^*} = 1$ and the limit $\lim_{\hat{u}} \to \infty$, $p(\hat{u})$ is a constant. The definition of projection operator is the same as that in Chapter 7.

With the additional system bounding information, the repetitive learning control achieves improved convergence property, as summarized in the following theorem.

Theorem 8.2. *For the system (8.1), under Assumption 8.1 and Assumption 8.2, the learning control law (8.6) and (8.20) guarantees the uniformly asymptotical convergence of* $\Delta\mathbf{x}$.

Proof. The solution $(\Delta\mathbf{x}, \hat{u})$ of the dynamic system (8.10) for $t \in [0, T)$ is the same as Theorem 8.1 *Part 1* without projection, because $\hat{u}(t-T) = 0$. To prove the existence of solution in $[T, \infty)$, define the same Lyapunov-Krasovskii functional in (8.15). The relations (8.16) and (8.12) still hold as the projection operation is not directly involved. Next look at the relation (8.18), which might be affected by the introduction of the projection operator.

We can easily verify the property $(u - \hat{u})^2 \geq [u - \mathcal{P}(\hat{u})]^2$, for any quantities $\hat{u}$. Using this property, the updating law (8.20), the periodic property $u_r = u_{r,\ominus}$, and the algebraic relation (8.17), we have

$$\begin{aligned} \frac{1}{2k_0}[(u_r - \hat{u})^2 - (u_{r,\ominus} - \hat{u}_\ominus)^2] &\leq \frac{1}{2k_0}[(u_r - \hat{u})^2 - (u_r - \mathcal{P}(\hat{u}_\ominus))^2] \\ &= \frac{1}{2k_0}[-2(u_r - \hat{u})(\hat{u} - \mathcal{P}(\hat{u}_\ominus)) - (\hat{u} - \mathcal{P}(\hat{u}_\ominus))^2] \\ &= \sigma\nu - \frac{k_0}{2}\sigma^2 \end{aligned}$$

which turns out to be the same as (8.18). In the sequel, the conclusion of *Part 2* in Theorem 8.1, namely the existence of solution $(\Delta\mathbf{x}, \hat{u})$ over the interval $[T, \infty)$, still holds. According to *Part 3* of Theorem 8.1, the integral convergence of $\Delta\mathbf{x}$, i.e.,

$$\lim_{t\to\infty} \int_{t-T}^{t} \|\Delta\mathbf{x}(\tau)\|^2 d\tau = 0$$

is obtained.

By virtue of the projection, the boundedness of $\Delta\mathbf{x}$ ensures the finiteness of $\hat{u}$, thereafter u and $\Delta\dot{\mathbf{x}}$. The boundedness of $\Delta\dot{\mathbf{x}}$ implies the uniform continuity of $\Delta\mathbf{x}$, therein the uniform continuity of the tracking error $\Delta\mathbf{x}$. As a result, $\lim_{t\to\infty} \|\Delta\mathbf{x}(t)\| = 0$. □

Remark 8.2. *In practice we may not know the exact value of the bound u^*. Instead we can choose u^* to be a sufficiently large constant. Note that u^* is used only as a saturator to limit the learning control effort, hence the controller gain will not be affected in the unsaturated region.*

8.4.2 Learning With Damping

When the bound u^* is not available, an alternative approach is the introduction of a damping (forgetting) factor. Note that the original updating law (8.7) is a pointwise integrator, that is, for any $t \in [(i-1)T,\ iT)$, it performs discrete-time integration over the time sequence $t - iT$ for $i = 1, 2, \cdots, i-1$. Such an integral mechanism might be sensitive to many non-ideal factors, such as biased measurement noise, the unmodeled higher order dynamics, etc. An effective modification is to add a "damping" term such that the parametric updating mechanism becomes a low pass filter instead of an integrator. As such, the updating law (8.7) can be modified as

$$\begin{aligned} \hat{u}(t) &= \gamma\hat{u}(t-T) - k(t)\sigma(t), \qquad (8.21) \\ \hat{u}(t) &= 0, \quad \forall t \in [-T,\ 0], \end{aligned}$$

where $0 < \gamma \le 1$ is the damping coefficient.

Different from projection, damping is introduced without using any extra system information. Hence it is a trade-off made between the robustness and the tracking convergence.

Theorem 8.3. *For system (8.1), under Assumption 8.1 and Assumption 8.2, the learning control law (8.6) and (8.21) guarantees the finiteness of the solution trajectory $(\Delta\mathbf{x}, \hat{u})$ in the large.*

Proof. The solution $(\Delta\mathbf{x}, \hat{u})$ for $t \in [0, T)$ is the same as Theorem 8.1 *Part 1* without damping, because $\hat{u}(t - T) = 0$. Thus in the following we discuss the solution in the interval $[T, \infty)$. Analogous to Theorem 8.1, assume the solution exists in $[T,\ (i-1)T)$ and can be continued up to $t_i \in [(i-1)T,\ iT)$. We need only to show the finiteness of the solution for any $t_i \in [(i-1)T,\ iT)$. Define the same Lyapunov-Krasovskii functional as (8.15) in Theorem 8.1. The relations (8.16) and (8.12) still hold as only the closed-loop dynamics is directly involved in the derivation. Next look at the relation (8.18), which is affected by the introduction of damping. Using the updating law (8.21), the periodic property $u_r = u_{r,\ominus}$ and the algebraic relation (8.17), we have

$$\begin{aligned}\frac{1}{2k_0}(\nu^2 - \nu_\ominus^2) &= \frac{1}{2k_0}[(u_r - \hat{u})^2 - (u_{r,\ominus} - \hat{u}_\ominus)^2] \\ &= \frac{1}{2k_0}[(u_r - \hat{u})^2 - (u_r - \hat{u}_\ominus)^2] \\ &= \frac{1}{2k_0}[-2(u_r - \hat{u})(\hat{u} - \hat{u}_\ominus) - (\hat{u} - \hat{u}_\ominus)^2] \\ &= -\frac{1}{k_0}(u_r - \hat{u})(\hat{u} - \gamma\hat{u}_\ominus) + \frac{1}{k_0}(1-\gamma)(u_r - \hat{u})\hat{u}_\ominus - \frac{1}{2k_0}(\hat{u} - \hat{u}_\ominus)^2.\end{aligned} \tag{8.22}$$

The first term on the right hand side of (8.22), by substituting the updating law (8.21), is $\sigma\nu$ which will cancel out the same term but with opposite sign in (8.12). In order to evaluate last two terms on the right hand side of (8.22), using the relationship $a^2 + b^2 \geq 2ab$, yields

$$\begin{aligned}&\frac{1}{k_0}(1-\gamma)(u_r - \hat{u})\hat{u}_\ominus - \frac{1}{2k_0}(\hat{u} - \hat{u}_\ominus)^2 \\ &= \frac{1}{k_0}(1-\gamma)(u_r\hat{u}_\ominus - \hat{u}\hat{u}_\ominus) - \frac{1}{2k_0}(\hat{u} - \hat{u}_\ominus)^2 \\ &\leq \frac{1}{2k_0}(1-\gamma)(u_r^2 + \hat{u}_\ominus^2 - 2\hat{u}\hat{u}_\ominus) - \frac{1}{2k_0}(\hat{u} - \hat{u}_\ominus)^2 \\ &\leq \frac{1}{2k_0}(1-\gamma)[u_r^2 - \hat{u}^2 + (\hat{u}^2 + \hat{u}_\ominus^2 - 2\hat{u}\hat{u}_\ominus)] - \frac{1}{2k_0}(\hat{u} - \hat{u}_\ominus)^2 \\ &\leq \frac{1-\gamma}{2k_0}(u_r^2 - \hat{u}^2) + \frac{1}{2k_0}(1-\gamma)(\hat{u} - \hat{u}_\ominus)^2 - \frac{1}{2k_0}(\hat{u} - \hat{u}_\ominus)^2 \\ &= \frac{1-\gamma}{2k_0}(u_r^2 - \hat{u}^2) - \frac{k_0\gamma}{2}\sigma^2.\end{aligned}$$

Therefore, the upper right hand derivative of V is

$$\begin{aligned}\dot{V} &\leq -\frac{\lambda_Q}{4}\|\Delta\mathbf{x}\|^2 - (\frac{\sqrt{\lambda_Q}}{2}\|\Delta\mathbf{x}\| - \frac{1}{\sqrt{\lambda_Q}}\alpha|\sigma|)^2 \\ &\quad -\frac{k_0\gamma}{2}\sigma^2 + \frac{1-\gamma}{2k_0}(u_r^2 - \hat{u}^2) \\ &\leq -\frac{\lambda_Q}{4}\|\Delta\mathbf{x}\|^2 - \frac{1-\gamma}{2k_0}\hat{u}^2 + \frac{1-\gamma}{2k_0}u_r^2 \end{aligned} \quad (8.23)$$

Now we can show the finiteness of V in the interval $[(i-1)T,\ t_i)$. If V is finite at $(i-1)T$, then it remains finite at t_i because $\dot{V}$ is uniformly bounded by $\frac{1-\gamma}{2k_0}\|u_r\|_s^2$. Consequently $\Delta\mathbf{x}$ and σ remain finite. The finiteness of $\hat{u}$ in the interval $[(i-1)T,\ t_i)$ can be derived from the finiteness of $\sigma(t)$ in (8.21). This implies the solution $(\Delta\mathbf{x}, \hat{u})$ either remains uniformly bounded or tend to infinity as $t \to \infty$. Thus the solution $(\Delta\mathbf{x}, \hat{u})$ exists for any $t \in [0,\ \infty)$.

We further show that the solution $(\Delta\mathbf{x}, \hat{u})$ remains finite when $t \to \infty$. From (8.23), $\dot{V} \leq 0$ as long as the solution $(\Delta\mathbf{x}, \hat{u})$ is outside a compact set $\mathcal{M}$ defined below

$$\mathcal{M} = \left\{(\Delta\mathbf{x}, \hat{u}) : \frac{\lambda_Q}{4}\|\Delta\mathbf{x}\|^2 + \frac{1-\gamma}{2k_0}|\hat{u}|^2 \leq \frac{1-\gamma}{2k_0}\|u_r\|_s^2\right\}.$$

where $M(\Delta\mathbf{x}, \hat{u}) \stackrel{\triangle}{=} \frac{\lambda_Q}{4}\|\Delta\mathbf{x}\|^2 + \frac{1-\gamma}{2k_0}|\hat{u}|^2$. Define an ϵ-neighborhood of $\mathcal{M}$ with $\epsilon > 0$

$$\mathcal{M}_\epsilon = \left\{(\Delta\mathbf{x}, \hat{u}) : \frac{\lambda_Q}{4}\|\Delta\mathbf{x}\|^2 + \frac{1-\gamma}{2k_0}|\hat{u}|^2 \leq \frac{1-\gamma}{2k_0}\|u_r\|_s^2 + \epsilon\right\},$$

then $\dot{V} \leq -\epsilon$ for any $(\Delta\mathbf{x}, \hat{u}) \in \mathcal{M}_\epsilon^c$ where $\mathcal{M}_\epsilon^c$ is the complementary set of $\mathcal{M}_\epsilon$. Since the solution exists in $[0,\ \infty)$, there is no finite escape time for $(\Delta\mathbf{x}, \hat{u})$. First assume that $\Delta\mathbf{x}$, thereby V, diverges asymptotically. Consider the fact that $\dot{V} \leq \frac{1-\gamma}{2k_0}\|u_r\|_s^2$, there must exist an infinite time interval $[t_s,\ \infty)$, such that

$$\frac{\lambda_Q}{4}\|\Delta\mathbf{x}\|^2 + \frac{1-\gamma}{2k_0}|\hat{u}|^2 \in \mathcal{M}_\epsilon^c \qquad \forall t \in [t_s,\ \infty).$$

Since the solution exists in $[0,\ \infty)$, $V(t_s, \Delta\mathbf{x}(t_s), \nu(t_s))$ is finite. Integrating $\dot{V}$ in (8.23) from $t \geq t_s$ we have

$$\lim_{t\to\infty} V(t) \leq V(t_s, \Delta\mathbf{x}(t_s), \nu(t_s)) - \lim_{t\to\infty}\int_{t_s}^{t} \epsilon d\tau \to -\infty,$$

that is however impossible because $V \geq 0$. We can conclude that $\Delta\mathbf{x}$ cannot stay infinitely long in $\mathcal{M}_\epsilon^c$, and will always re-enter $\mathcal{M}_\epsilon$ after a finite interval. Hence $\Delta\mathbf{x}$ remains finite when $t \to \infty$. Note that the finiteness of $\Delta\mathbf{x}$ warrants the finiteness of $\sigma(t)$ over the entire horizon $[0, \infty)$. On the other hand, the learning law (8.21) with the damping γ is an asymptotically stable first order difference equation subject to the input $k(t)\sigma(t)$. Therefore $\hat{u}$ remains finite when $t \to \infty$. □

8.5 RLC Extensions

We consider two extensions: the first is an extension to the system (8.1) with unknown input coefficient, and the second is an extension to a cascaded dynamics with unmatched components.

8.5.1 Plant with Unknown Input Coefficient

Consider a specific case below

$$\begin{aligned} \dot{x}_j &= x_{j+1}, & j = 1, 2, \cdots, n-1, \\ \dot{x}_n &= \eta(t, \mathbf{x}) + b(t, \mathbf{x})u(t), & \mathbf{x}(0) = \mathbf{x}_0. \end{aligned} \tag{8.24}$$

If $b(t, \mathbf{x})$ is known and nonsingular, the RLC can be constructed directly by multiplying the robust control part with the factor $b^{-1}(t, \mathbf{x})$. In the following we focus on the case that $b(t, \mathbf{x}) = b$ is a constant with a known lower bound b_{min}. Without loss of generality, assume $b \geq b_{min} > 0$. Note that the presence of the constant input coefficient b does not change the periodicity of the ideal control obtainable from the following dynamic relationship

$$\dot{x}_{r,n}(t) = \eta(t, \mathbf{x}_r) + bu_r(t).$$

Hence the proposed repetitive learning control approach is still applicable.

However, the robust control part will have to be revised. It is worth to point out that the lower bound b_{min} is required by most existing robust control methods which however may not be able to cope with the system (8.24) due to the lumped uncertain component $\eta(t, \mathbf{x})$ under Assumption 8.2. Let us derive the robust control part. From (8.24), the tracking error dynamics is

$$\begin{aligned} \Delta\dot{\mathbf{x}} &= A\Delta\mathbf{x} + \mathbf{b}(\mathbf{c}\Delta\mathbf{x} + \eta - \eta_r + bu - bu_r) \\ &= A\Delta\mathbf{x} + b\mathbf{b}[b^{-1}\mathbf{c}\Delta\mathbf{x} + b^{-1}(\eta - \eta_r) + u - u_r]. \end{aligned}$$

Because of the unknown input coefficient b, $\mathbf{c}\Delta\mathbf{x}$ cannot be compensated directly by the control input u. Instead, we can treat $b^{-1}\mathbf{c}\Delta\mathbf{x} + b^{-1}(\eta - \eta_r)$ as a lumped uncertainty with an upper bound on the variation which, referring to Assumption 8.2, is

$$\bar{\alpha} = \frac{1}{b_{min}}(\|\mathbf{c}\| + \alpha).$$

Accordingly the revised learning control law is

$$\begin{aligned} u(t) &= \hat{u}(t) - \frac{1}{\lambda_Q d_{\min}}\bar{\alpha}^2\sigma(t) \\ \hat{u}(t) &= \hat{u}(t-T) - k(t)\sigma(t). \end{aligned} \tag{8.25}$$

The Lyapunov-Krasovskii functional is chosen to be

$$V(t, \Delta\mathbf{x}, \nu) = \frac{1}{2b}\Delta\mathbf{x}^T P \Delta\mathbf{x} + \frac{1}{2k_0}\int_{t-T}^{t} \nu^2 d\tau. \tag{8.26}$$

The upper right hand derivative is

$$\dot{V} = \frac{1}{2b}(\Delta\dot{\mathbf{x}}^T P\Delta\mathbf{x} + \Delta\mathbf{x}^T P\Delta\dot{\mathbf{x}}) + \frac{1}{2k_0}(\nu^2 - \nu_{\ominus}^2). \tag{8.27}$$

It can be seen from the new learning control law (8.25), the Lyapunov-Krasovskii functional V in (8.26), and its derivative $\dot{V}$ in (8.27) that all terms related to $\hat{u}$ and $\hat{u} - u_r$ remain the same as the preceding case in Theorem 8.1. Thus we need only to evaluate the first term, $\frac{1}{2b}(\Delta\dot{\mathbf{x}}^T P\Delta\mathbf{x} + \Delta\mathbf{x}^T P\Delta\dot{\mathbf{x}})$, on the right hand side of (8.27) as it is affected directly by the unknown input coefficient. Notice the fact $1/b_{\min} \geq 1/b$, we have

$$\begin{aligned}
& \frac{1}{2b}(\Delta\dot{\mathbf{x}}^T P \Delta\mathbf{x} + \Delta\mathbf{x}^T P \Delta\dot{\mathbf{x}}) \\
\leq & -\frac{\lambda_Q}{2b}\|\Delta\mathbf{x}(t)\|^2 + \sigma[b^{-1}\mathbf{c}\Delta\mathbf{x} + b^{-1}(\eta - \eta_r) - \frac{1}{\lambda_Q b_{\min}}\bar{\alpha}^2\sigma] - \sigma\nu \\
\leq & -\frac{\lambda_Q}{2b}\|\Delta\mathbf{x}(t)\|^2 + b^{-1}\bar{\alpha}|\sigma| \cdot \|\Delta\mathbf{x}\| - \frac{1}{\lambda_Q b}\bar{\alpha}^2\sigma^2 - \sigma\nu \\
\leq & -\frac{\lambda_Q}{4b}\|\Delta\mathbf{x}(t)\|^2 - \sigma\nu - \frac{1}{b}(\frac{\sqrt{\lambda_Q}}{2}\|\Delta\mathbf{x}\| - \frac{1}{\sqrt{\lambda_Q}}\bar{\alpha}|\sigma|)^2
\end{aligned} \tag{8.28}$$

Clearly, (8.28) has the similar form as (8.12) except for the extra coefficient b which however does not change the negativeness property of the first two terms on the right hand side of (8.28). As a result, all the derivations and the convergence property in the proof of Theorem 8.1 still hold.

8.5.2 Plant in Cascaded Form

Consider the following n-th order cascaded dynamic system

$$\begin{aligned}
\dot{x}_j &= x_{j+1} + \eta_1(t, \mathbf{x}_j), \\
\dot{x}_n &= u + \eta_n(t, \mathbf{x}),
\end{aligned} \tag{8.29}$$

where $\mathbf{x}_j = [x_1, \cdots, x_j]^T$, $\mathbf{x} = \mathbf{x}_n$, and $\eta_j(t, \mathbf{x}_j)$ are nonlinear unknown functions continuously differentiable w.r.t the arguments t and $\mathbf{x}_j$. Here η_j ($j = 1, \cdots, n-1$) are unmatched uncertainties. The backstepping design has been developed as a systematic approach to handle cascaded dynamics or any systems in triangular form. The principal idea of backstepping design is for the i-th subsystem to construct a fictitious control input, which will enter the $(i+1)$-th subsystem as the objective trajectory and will be differentiated. In RLC, however, the learning updating law (8.7) is a continuous-time difference equation, differentiating it leads to

$$\dot{\hat{u}}(t) = \dot{\hat{u}}(t - T) - \dot{k}(t)\sigma(t) - k(t)\dot{\sigma}(t).$$

It requires the derivative signals of $\hat{u}$, which are obviously unavailable in practice.

In what follows we will demonstrate how is the repetitive learning integrated with robust adaptation to facilitate the backstepping design. As a systematic method, the backstepping design can be easily extended from second order to n-th order, hence for simplicity we consider a second order dynamics, i.e. $n = 2$ in (8.29), so as to concentrate on the most fundamental steps in the problem solving.

The control objective is to design an appropriate control input $u(t)$ such that x_1 can track $x_{r,1}$ that is generated by the reference model (8.2). The reference trajectory $\mathbf{x}_r(t)$, and the quantity $\eta_1(t, x_{r,1})$ and $\eta_2(t, x_{r,1}, x_{r,2})$ satisfy Assumption 1, i.e., $\mathbf{x}_r(t) \in \mathcal{C}^2_{PT}([0,\infty); \mathcal{R}^2)$, $\eta_1(t, x_{r,1}) = \eta_1(t-T, x_{r,1})$ and $\eta_2(t, \mathbf{x}_r) = \eta_2(t-T, \mathbf{x}_r)$. Furthermore, $\eta_1(t, x_1)$ and $\eta_2(t, \mathbf{x})$ satisfy Assumption 2, i.e.,

$$|\eta_1(t,x) - \eta_1(t,y)| \le \alpha_1(t,x,y)\|x-y\|,$$

and

$$|\eta_2(t,\mathbf{x}) - \eta_2(t,\mathbf{y})| \le \alpha_2(t,\mathbf{x},\mathbf{y})\|\mathbf{x}-\mathbf{y}\|,$$

where $\alpha_1(t,x,y)$ and $\alpha_2(t,\mathbf{x},\mathbf{y})$ are known bounding functions.

For notational convenience, in subsequent context, we denote $\eta_1 \triangleq \eta_1(t,x_1)$, $\eta_2 \triangleq \eta_2(t,\mathbf{x})$, $\eta_{r,1} \triangleq \eta_1(t,x_{r,1})$, $\eta_{r,2} \triangleq \eta_2(t,\mathbf{x}_r)$, and $\alpha_1 \triangleq \alpha_1(t,x_1,x_{r,1})$. Specifically, denote $\alpha_2 \triangleq \alpha_2(t,\mathbf{x},\mathbf{y})$ when $\mathbf{x} = [x_1, x_2]^T$ and $\mathbf{y} = [x_{r,1}, x_2]^T$, and denote $\alpha_2' \triangleq \alpha_2(t,\mathbf{x},\mathbf{y})$ when $\mathbf{x} = [x_{r,1}, x_2]^T$ and $\mathbf{y} = [x_{r,1}, x_{r,2}]^T$.

It is obvious that $\eta_{r,j} \in \mathcal{C}^1_{PT}([0,\infty); \mathcal{R}^1)$, $j = 1,2$, thus will be learned. On the other hand, $\eta_{r,1}$ is finite, though the upper bound is unknown to us. Let β denote the upper bound of $\eta_{r,1}$.

Denote

$$\mathcal{S}(x) = k_1 arctan(k_2 x), \tag{8.30}$$

for any variable x, where $k_1 > 0$ and $k_2 > 0$ are design parameters. Note that if choosing gains k_1 and k_2 such that

$$\frac{1}{k_2} tan \frac{1}{k_1} \leq \delta,$$

then

$$x\mathcal{S}(x) = xk_1 arctan(k_2 x) \geq \begin{cases} |x| & |x| \geq \delta \\ x^2/\delta & |x| < \delta, \end{cases} \tag{8.31}$$

It is easy to verify that $\mathcal{S}(x)$ is continuously differentiable and possessing the following property.

Property 8.1.

$$|x| - \mathcal{S}(x)x \leq \delta.$$

Proof. From the definition of $\mathcal{S}(x)$, it is easy to have $|x| - \mathcal{S}(x)x \leq 0 < \delta$ for $|x| \geq \delta$. For $|x| < \delta$, we have

$$|x| - \mathcal{S}(x)x \leq |x| - x^2/\delta \leq |x| \leq \delta.$$

Thus the result holds. □

Define new coordinates $z_1 = x_1 - x_{r,1}$ and $z_2 = x_2 - u_1$, where the fictitious control is

$$u_1 = -(\alpha_1 + q_1)z_1 + x_{r,2} - \mathcal{S}(\hat{\beta} z_1)\hat{\beta} \tag{8.32}$$

with $q_1 > 0$. $\hat{\beta}$ is the estimation of β

$$\dot{\hat{\beta}} = |z_1| - \gamma\hat{\beta}, \tag{8.33}$$

where $\gamma > 0$ is a damping coefficient.

Design the actual controller

$$u = f_2 - z_1 - q_2 z_2 - \mathcal{S}(\bar{\alpha}_2 z_2)\bar{\alpha}_2 - \hat{\boldsymbol{\theta}}^T \boldsymbol{\xi} \tag{8.34}$$

with $q_2 > 0$, $\boldsymbol{\xi} = [-\frac{\partial u_1}{\partial x_1} \quad 1]^T$,

$$f_2 = \frac{\partial u_1}{\partial t} + \frac{\partial u_1}{\partial x_{r,1}} x_{r,2} + \frac{\partial u_1}{\partial x_{r,2}} s(t, \mathbf{x}_r, r) + \frac{\partial u_1}{\partial \hat{\beta}} \dot{\hat{\beta}} + \frac{\partial u_1}{\partial x_1} x_2,$$

and

$$\bar{\alpha}_2 = \left(\alpha_2 + \alpha_1 \left| \frac{\partial u_1}{\partial x_1} \right| \right) |\Delta x_1| + \alpha_2^{'} |\Delta x_2|.$$

$\hat{\boldsymbol{\theta}}$ is to learn $\boldsymbol{\theta} = [\eta_{r,1} \quad \eta_{r,2}]^T$ which are periodic. The learning law is

$$\hat{\boldsymbol{\theta}} = \hat{\boldsymbol{\theta}}_\ominus + \boldsymbol{\xi} z_2, \tag{8.35}$$

where $\hat{\boldsymbol{\theta}}_\ominus = \hat{\boldsymbol{\theta}}(t - T)$.

Theorem 8.4. *For system (8.29), the control law (8.34), the adaptation law (8.33) and learning law (8.35) guarantee the finiteness of z_1 and z_2 in the large, and the tracking error bound of z_1 is*

$$|z_1| \leq \sqrt{\frac{4\delta + \gamma\beta^2}{2q_1}}. \tag{8.36}$$

Proof. The proof consists of two steps.

Step 1.

From (8.29) and (8.2), we have

$$\begin{aligned} \dot{z}_1 &= \dot{x}_1 - \dot{x}_{r,1} \\ &= x_2 + \eta_1 - x_{r,2} \\ &= z_2 + u_1 + \eta_1 - x_{r,2}. \end{aligned} \tag{8.37}$$

Substituting the fictitious control u_1 (8.32) into (8.37) yields

$$\begin{aligned} \dot{z}_1 &= z_2 - (\alpha_1 + q_1) z_1 + \eta_1 - \mathcal{S}(\hat{\beta} z_1)\hat{\beta} \\ &= z_2 - (\alpha_1 + q_1) z_1 - \mathcal{S}(\hat{\beta} z_1)\hat{\beta} + (\eta_1 - \eta_{r,1}) + \eta_{r,1}. \end{aligned} \tag{8.38}$$

Define a Lyapunov function candidate below

$$V_1 = \frac{1}{2} z_1^2 + \frac{1}{2} (\beta - \hat{\beta})^2. \tag{8.39}$$

Using (8.38), adaptation law (8.33) and Property 8.1, the derivative of V_1 is

$$\begin{aligned}
\dot{V}_1 &= z_1\dot{z}_1 - (\beta-\hat{\beta})\dot{\hat{\beta}} \\
&= z_1[z_2 - (\alpha_1+q_1)z_1 - \mathcal{S}(\hat{\beta}z_1)\hat{\beta} + (\eta_1-\eta_{r,1}) + \eta_{r,1}] - (\beta-\hat{\beta})\dot{\hat{\beta}} \\
&= z_1z_2 - (\alpha_1+q_1)z_1^2 + (\eta_1-\eta_{r,1})z_1 - \mathcal{S}(\hat{\beta}z_1)\hat{\beta}z_1 + \eta_{r,1}z_1 - (\beta-\hat{\beta})\dot{\hat{\beta}} \\
&\le z_1z_2 - q_1z_1^2 - \mathcal{S}(\hat{\beta}z_1)\hat{\beta}z_1 + \beta|z_1| - (\beta-\hat{\beta})\dot{\hat{\beta}} \\
&= z_1z_2 - q_1z_1^2 - \mathcal{S}(\hat{\beta}z_1)\hat{\beta}z_1 + \hat{\beta}|z_1| - \hat{\beta}|z_1| + \beta|z_1| - (\beta-\hat{\beta})\dot{\hat{\beta}} \\
&\le z_1z_2 - q_1z_1^2 + |\hat{\beta}z_1|[1-|\mathcal{S}(\hat{\beta}z_1)|] - (\beta-\hat{\beta})(\dot{\hat{\beta}} - |z_1|) \\
&\le z_1z_2 - q_1z_1^2 + \delta - (\beta-\hat{\beta})(\dot{\hat{\beta}} - |z_1|).
\end{aligned} \tag{8.40}$$

Step 2.

From (8.29) and (8.32), we have

$$\begin{aligned}
\dot{z}_2 &= \dot{x}_2 - \dot{u}_1 \\
&= u + \eta_2 - \left(\frac{\partial u_1}{\partial t} + \frac{\partial u_1}{\partial x_1}\dot{x}_1 + \frac{\partial u_1}{\partial x_{r,1}}x_{r,2} + \frac{\partial u_1}{\partial x_{r,2}}s(t,\mathbf{x}_r,r) + \frac{\partial u_1}{\partial \hat{\beta}}\dot{\hat{\beta}}\right) \\
&= u - \left(\frac{\partial u_1}{\partial t} + \frac{\partial u_1}{\partial x_{r,1}}x_{r,2} + \frac{\partial u_1}{\partial x_{r,2}}s(t,\mathbf{x}_r,r) + \frac{\partial u_1}{\partial \hat{\beta}}\dot{\hat{\beta}}\right) + \eta_2 - \frac{\partial u_1}{\partial x_1}(x_2+\eta_1) \\
&= u - f_2 - \frac{\partial u_1}{\partial x_1}\eta_{r,1} + \eta_{r,2} \\
&\quad -\frac{\partial u_1}{\partial x_1}(\eta_1-\eta_{r,1}) + [\eta_2 - \eta_2(t,x_{r,1},x_2)] + [\eta_2(t,x_{r,1},x_2) - \eta_{r,2}] \\
&= u - f_2 + \boldsymbol{\theta}^T\boldsymbol{\xi} - \frac{\partial u_1}{\partial x_1}(\eta_1-\eta_{r,1}) + [\eta_2 - \eta_2(t,x_{r,1},x_2)] + [\eta_2(t,x_{r,1},x_2) - \eta_{r,2}],
\end{aligned} \tag{8.41}$$

where

$$\begin{aligned}
f_2 &= \frac{\partial u_1}{\partial t} + \frac{\partial u_1}{\partial x_{r,1}}x_{r,2} + \frac{\partial u_1}{\partial x_{r,2}}s(t,\mathbf{x}_r,r) + \frac{\partial u_1}{\partial \hat{\beta}}\dot{\hat{\beta}} + \frac{\partial u_1}{\partial x_1}x_2, \\
\boldsymbol{\xi} &= [-\frac{\partial u_1}{\partial x_1} \quad 1]^T,
\end{aligned}$$

are known and

$$\boldsymbol{\theta} = [\eta_{r,1} \quad \eta_{r,2}]^T$$

is to be learned.

Substituting (8.34) into (8.41) yields

$$\begin{aligned}\dot{z}_2 &= -z_1 - q_2 z_2 + (\boldsymbol{\theta} - \hat{\boldsymbol{\theta}})^T \boldsymbol{\xi} - \mathcal{S}(\bar{\alpha}_2 z_2)\bar{\alpha}_2 \\ &\quad -\frac{\partial u_1}{\partial x_1}(\eta_1 - \eta_{r,1}) + [\eta_2 - \eta_2(t, x_{r,1}, x_2)] + [\eta_2(t, x_{r,1}, x_2) - \eta_{r,2}] \end{aligned} \quad (8.42)$$

Define the Lyapunov functional below

$$V_2 = V_1 + \frac{1}{2} z_2^2 + \frac{1}{2}\int_{t-T}^{t} (\boldsymbol{\theta} - \hat{\boldsymbol{\theta}})^T (\boldsymbol{\theta} - \hat{\boldsymbol{\theta}}) d\tau. \quad (8.43)$$

The upper right hand derivative of V_2 is

$$\begin{aligned}\dot{V}_2 &= \dot{V}_1 + z_2\dot{z}_2 + \frac{1}{2}(\boldsymbol{\theta} - \hat{\boldsymbol{\theta}})^T(\hat{\boldsymbol{\theta}} - \hat{\boldsymbol{\theta}}) - \frac{1}{2}(\hat{\boldsymbol{\theta}} - \hat{\boldsymbol{\theta}}_\ominus)^T(\hat{\boldsymbol{\theta}} - \hat{\boldsymbol{\theta}}_\ominus) \\ &\leq \dot{V}_1 + z_2\dot{z}_2 - (\boldsymbol{\theta} - \hat{\boldsymbol{\theta}})^T(\hat{\boldsymbol{\theta}} - \hat{\boldsymbol{\theta}}_\ominus) \end{aligned} \quad (8.44)$$

where the last term on the right hand side is derived by using the algebraic relation (8.17) in vector form $(\mathbf{a}-\mathbf{b})^T(\mathbf{a}-\mathbf{b})-(\mathbf{a}-\mathbf{c})^T(\mathbf{a}-\mathbf{c}) = -2(\mathbf{a}-\mathbf{b})^T(\mathbf{b}-\mathbf{c})-\|\mathbf{b}-\mathbf{c}\|^2$.

Using (8.42) and Property 8.1, we have

$$\begin{aligned} z_2\dot{z}_2 &= -z_1 z_2 - q_2 z_2^2 + (\boldsymbol{\theta} - \hat{\boldsymbol{\theta}})^T \boldsymbol{\xi} z_2 - \mathcal{S}(\bar{\alpha}_2 z_2)\bar{\alpha}_2 z_2 \\ &\quad -\frac{\partial u_1}{\partial x_1}(\eta_1 - \eta_{r,1}) z_2 + [\eta_2 - \eta_2(t, x_{r,1}, x_2)] z_2 + [\eta_2(t, x_{r,1}, x_2) - \eta_{r,2}] z_2 \\ &\leq -z_1 z_2 - q_2 z_2^2 + (\boldsymbol{\theta} - \hat{\boldsymbol{\theta}})^T \boldsymbol{\xi} z_2 + \bar{\alpha}_2 |z_2| - \mathcal{S}(\bar{\alpha}_2 z_2)\bar{\alpha}_2 z_2 \\ &\leq -z_1 z_2 - q_2 z_2^2 + \boldsymbol{\xi}^T(\boldsymbol{\theta} - \hat{\boldsymbol{\theta}}) z_2 + \delta \end{aligned} \quad (8.45)$$

Substituting (8.40) and (8.45) into (8.44) yields

$$\begin{aligned}\dot{V}_2 &\leq -q_1 z_1^2 - q_2 z_2^2 + 2\delta - (\beta - \hat{\beta})(\dot{\hat{\beta}} - |z_1|) \\ &\quad -(\boldsymbol{\theta} - \hat{\boldsymbol{\theta}})^T(\hat{\boldsymbol{\theta}} - \hat{\boldsymbol{\theta}}_\ominus - \boldsymbol{\xi} z_2) \end{aligned} \quad (8.46)$$

Note the adaptation law (8.33) and learning law (8.35), we have

$$\begin{aligned}\dot{V}_2 &\leq -q_1 z_1^2 - q_2 z_2^2 + 2\delta + \gamma\hat{\beta}(\beta - \hat{\beta}) \\ &\leq -q_1 z_1^2 - q_2 z_2^2 + 2\delta - \gamma(\frac{1}{2}\hat{\beta}^2 - \hat{\beta}\beta) \\ &= -q_1 z_1^2 - q_2 z_2^2 - \frac{\gamma}{2}(\hat{\beta} - \beta)^2 + 2\delta + \frac{\gamma}{2}\beta^2 \end{aligned} \quad (8.47)$$

$\dot{V}_2$ is negative definite outside the compact set

$$\mathcal{M} = \{(z_1, z_2) : q_1 z_1^2 + q_2 z_2^2 + \frac{\gamma}{2}(\hat{\beta} - \beta)^2 \leq 2\delta + \frac{\gamma_1}{2}\beta^2\}.$$

Further define $\epsilon-$neighborhood of $\mathcal{M}$ with $\epsilon > 0$

$$\mathcal{M}_\epsilon = \{(z_1, z_2) : q_1 z_1^2 + q_2 z_2^2 + \frac{\gamma}{2}(\hat{\beta} - \beta)^2 \leq 2\delta + \frac{\gamma_1}{2}\beta^2 + \epsilon\}, \tag{8.48}$$

then $\dot{V}_2 \leq -\epsilon$. The state z_1 will enter the $\epsilon-$neighborhood, $\mathcal{M}_\epsilon$, in finite time, which implies the asymptotic convergence to the region (8.36). □

Remark 8.3. *From (8.48), it is clear that the size of $\mathcal{M}_\epsilon$ is decided by the design parameters q_1, q_2, δ and γ. Therefore the tracking error can be made sufficiently small by choosing appropriate values for the design parameters.*

Remark 8.4. *Adaptive robust control method can also be applied to dealing with the terms $\frac{\partial u_1}{\partial x_1}\eta_{r_1}$ and η_{r_2} in the second step. Differing from repetitive learning control used in the above, it will bring a high gain in the control law. Adaptive robust control design is given in Appendix A.6.*

Remark 8.5. *Though only second order cascaded system is considered, the results can be extended straightforward to n-th order cascaded systems.*

Remark 8.6. *The preceding robusitification schemes can also be applied to the repetitive learning law (8.35).*

8.6 Illustrative Examples

In this section, two illustrative examples are given for nonlinear systems with matched and unmatched uncertainties respectively. For simplicity the control performance is evaluated using the maximum absolute tracking error over one period T, denoted by MAE_T.

8.6.1 Nonlinear system with matched uncertainties

Consider a second order system described by (8.1) with matched uncertainties. Following the design procedure (8.4), choose $\mathbf{c} = [1,\ 1]$, then $A = \begin{bmatrix} 0 & 1 \\ -1 & -1 \end{bmatrix}$. Choosing $Q = I_{2\times 2}$ to be an identity matrix, the solution of the Lyapunov equation is $P = \begin{bmatrix} 1.5 & 0.5 \\ 0.5 & 1 \end{bmatrix}$. Choose $k_1(t) = k_0(-\frac{2}{T^3}t^3 + \frac{3}{T^2}t^2)$, which is smooth and monotone between 0 and $k_0 = 4$.

<u>*Case 1*</u>:

In the system (8.1), assume the lumped unknown is $\eta(t,\mathbf{x}) = (1+\sin x_2)x_1^2$. The reference model (8.2) is

$$\begin{aligned} \dot{x}_{r,1} &= x_{r,2}, \\ \dot{x}_{r,2} &= \sin \pi t \end{aligned}$$

with the initial values $\mathbf{x}_r(0) = [0,\ -\frac{1}{\pi}]$. The learning period thus is $T = 2$.

The known bounding function is chosen to be $\alpha(t,\mathbf{x},\mathbf{x}_r) = \sqrt{\alpha_1^2(\mathbf{x},t) + \alpha_2^2(\mathbf{x}_r,t)}$, where $\alpha_1(t,\mathbf{x}) = \sqrt{4x_1^2 + x_1^4\cos^2 x_2}$ and $\alpha_2(t,\mathbf{x}_r) = \sqrt{4x_{r,1}^2 + x_{r,1}^4\cos^2 x_{r,2}}$. Initial values are $\mathbf{x}(0) = [1,\ 0]$. Applying the repetitive learning control law (8.6) and (8.7), the learning convergence results of the tracking error and control profile are shown in Figure 8.1 and Figure 8.2 respectively. It is worthwhile highlighting that the learnt control $\hat{u}$ approaches the ideal one, in the sequel the robust control part will die out accordingly.

<u>*Case 2*</u>:

Assume that there exists an unmodeled dynamics – a second order resonance mode,

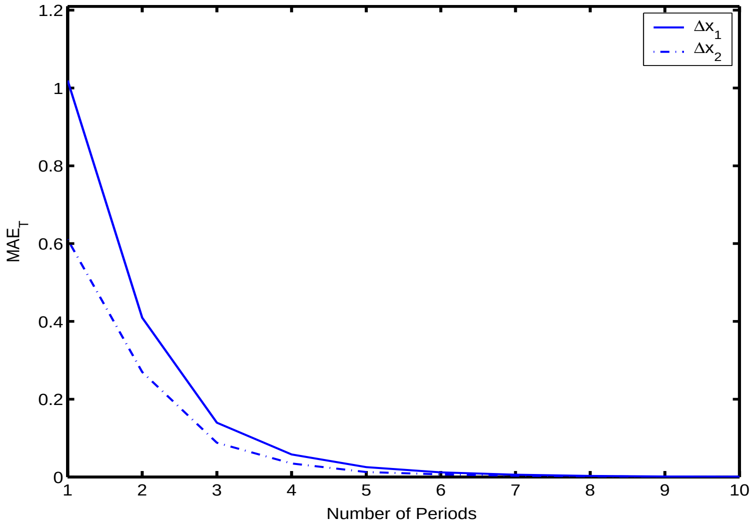

Figure 8.1. Learning convergence of the tracking errors (Case 1)

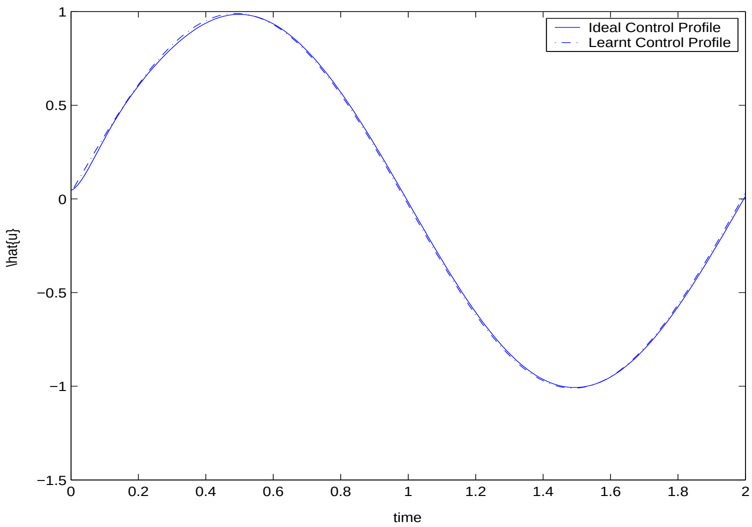

Figure 8.2. Ideal and learned control profiles at 10th period (Case 1)

and the actual plant is

$$\begin{aligned}\dot{x}_1 &= x_2,\\ \dot{x}_2 &= -30x_2 - 229x_1 + 229x_3,\\ \dot{x}_3 &= x_4,\\ \dot{x}_4 &= (1+\sin x_4)x_3^2 + u.\end{aligned}$$

The unmodeled dynamics is seen to have the transfer function relation $229/(s^2 +$

$30s + 229$). This is analogous to the well known example (Rohrs *et al.*, 1985) in adaptive control that is used to demonstrate the parameter drifting problem, thereby the necessary of robust modification.

Since the unmodeled dynamics is unknown to us, advanced control design methods such as backstepping method cannot be applied. Although x_3 and x_4 should be used in the control implementation, the actual control implementation is accomplished with only x_1 and x_2 which are the actual system output and its variation.

The result of RLC without any robustification is shown in Figure 8.3. It can be

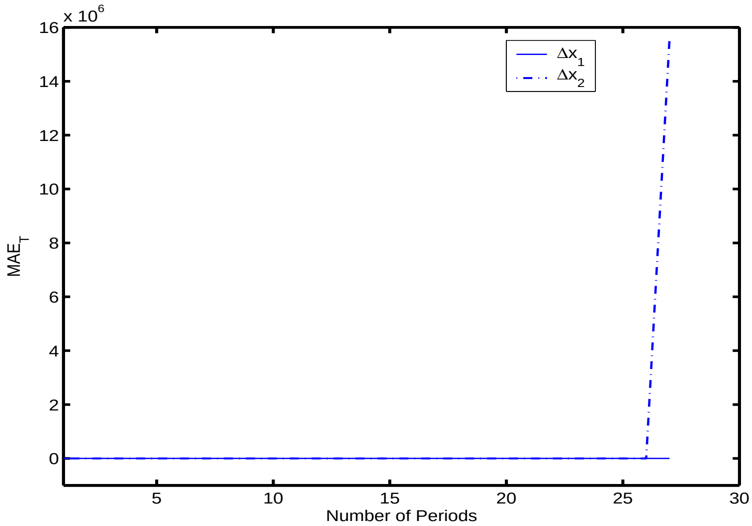

Figure 8.3. Tracking errors with unmodeled dynamics (Case 2)

seen that the tracking error Δx_2 diverges at the 27-th period.

Now RLC with projection is applied. The bound of $u_r(t)$ is assumed to be 3. The simulation result is shown in Figure 8.4. It can be observed that RLC with projection improves the performance greatly.

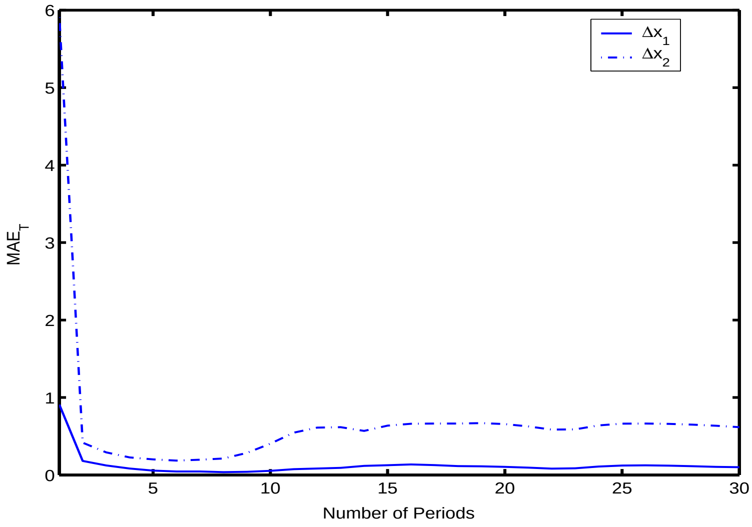

Figure 8.4. Tracking errors with unmodeled dynamics and learning projection (Case 2)

8.6.2 Nonlinear system with unmatched uncertainties

Now consider the following cascade dynamic system

$$\begin{aligned} \dot{x}_1 &= x_2 + \log(2 + x_1^2), \\ \dot{x}_2 &= u - \sqrt{6400 + x_1^2} - 10 \sin 5\pi t. \end{aligned} \tag{8.49}$$

with the initial values $\mathbf{x}(0) = [0.5 \quad 1]^T$. The desired target is $x_{r,1} = \frac{1}{25}\cos 5\pi t + 3$, and the learning period is $T = 0.4$.

The known variation bounding functions are $\alpha_1 = \sqrt{x_1^2 + 16}$, and $\alpha_2 = \alpha_2' = 1$, respectively. Let $q_1 = q_2 = 2$, $\gamma = 0.001$ and $\delta = 0.01$. Choose $k_1 = 1$ and $k_2 = 156$ such that $\frac{1}{k_2} tan \frac{1}{k_1} \leq \delta$. Applying the integrated control law (8.34), (8.33) and (8.35), the tracking error and control profiles are give in Figure 8.5 and Figure 8.6 respectively. The control profiles of the learning part are given in Figure 8.7.

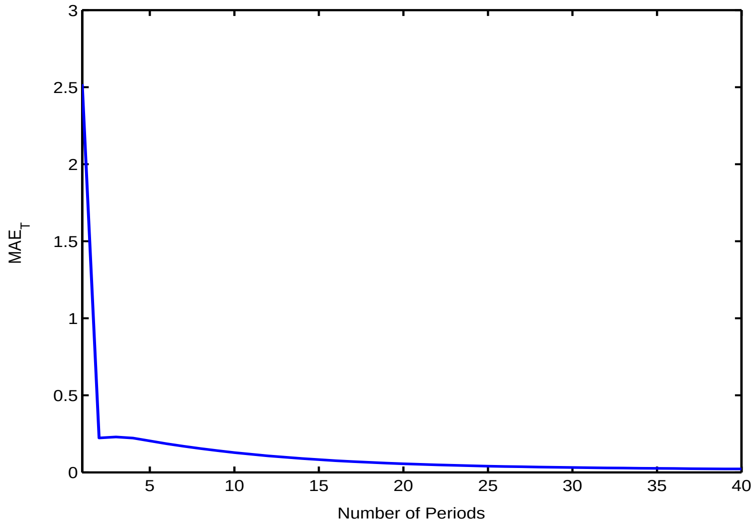

Figure 8.5. Tracking error z_1 with unmatched uncertainties

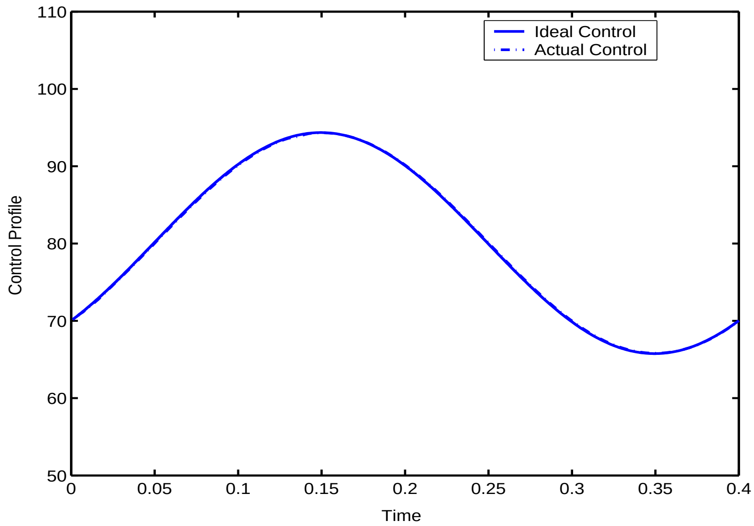

Figure 8.6. Ideal and actual control profiles at 40th period

According to Theorem 8.4, the upper bound of the tracking error z_1 is

$$
\begin{aligned}
|z_1| &\leq \sqrt{\frac{4\delta + \gamma\beta^2}{2q_1}} \\
&\leq \sqrt{\frac{4 \times 0.01 + 0.001 \log^2(2+16)}{4}} \\
&= 0.1099.
\end{aligned}
$$

Clearly, the simulation result is consistent with the conclusion in Theorem 8.4. We can also observe the convergence of the real control input to the ideal one with the

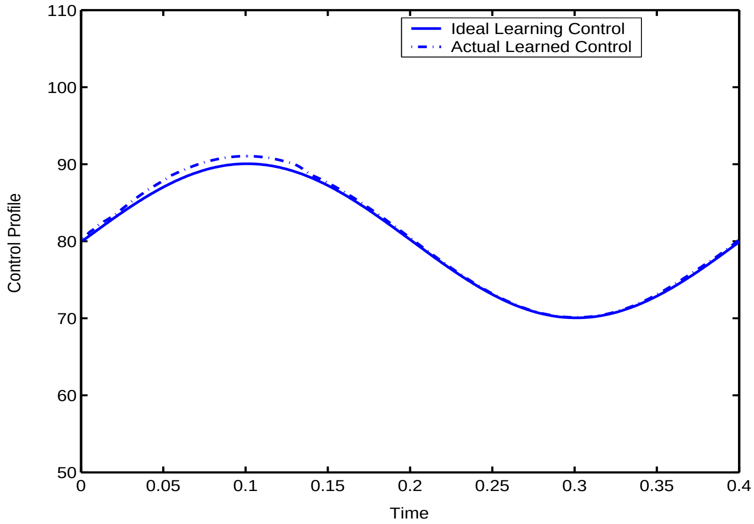

Figure 8.7. Ideal and actual learning control components at 40th period

learning and adaptation.

For comparison purpose the adaptive robust control method is also applied. The upper bound of tracking error z_1 is

$$\begin{aligned} |z_1| &\leq \sqrt{\frac{6\delta + \gamma_1\beta_1^2 + \gamma_2\beta_2^2}{2q_1}} \\ &\leq \sqrt{\frac{6 \times 0.01 + 0.001 \log^2(2+16) + 0.001 \times 91}{4}} \\ &\leq 0.1867. \end{aligned} \tag{8.50}$$

Case 1 Choosing the same design parameters as the repetitive learning control method. The actual tracking error is 0.0082 at the second period. Clearly the adaptive robust control method is a conservative design. Figure 8.8 and Figure 8.9 display the actual control profile and the adaptive robust part of control profile at 2th period respectively. Due to the conservative nature, high feedback gains are used, leading to extremely large control profiles. From Figure 8.9, the divergent trend of the adaptive robust control signals can be observed. In fact, all simulations in this chapter were conducted using the Runge Kutta 4-5th order with variable step size, and the controllers are simulated as continuous ones This implies that the preceding adaptive robust control design is not suitable for any digital implementation.

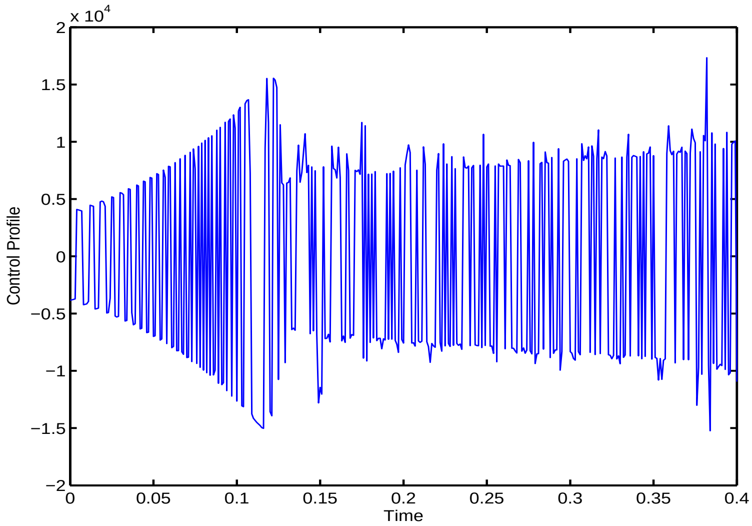

Figure 8.8. Actual control profile at 2th period

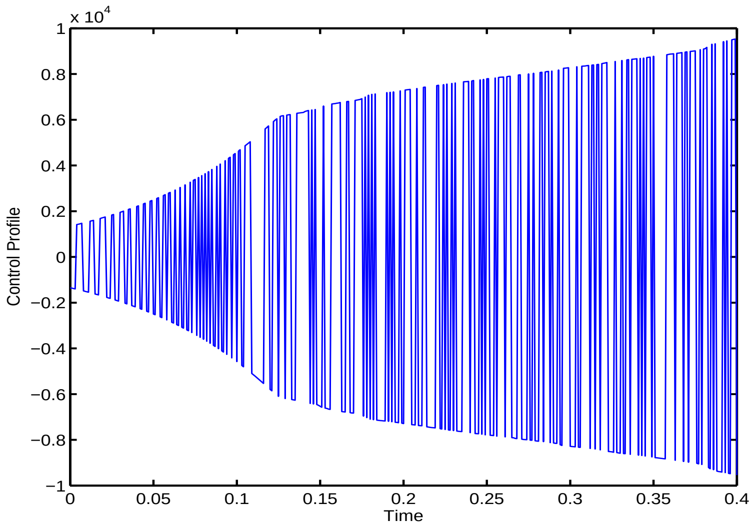

Figure 8.9. Adaptive robust part of the control profile at 2th period

Case 2 To mitigate the conservativeness of the adaptive robust controller, choose $k_1 = 1$ and $k_2 = 10$ such that $\delta \approx 1.56$ and the theoretically guaranteed error bound is 1.5378. Let other parameters be the same as the preceding case. The tracking error is given in Figure 8.10. The control signals is shown in Figure 8.11. From Figure 8.11, we can see that the actual control signal converges to the ideal control signal after 2th period using low gain feedback. Clearly, ARC is a conservative

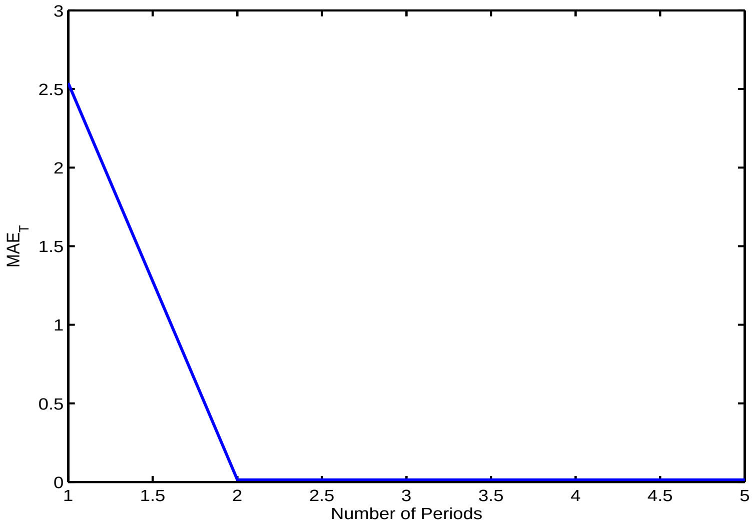

Figure 8.10. Tracking error z_1 with ARC

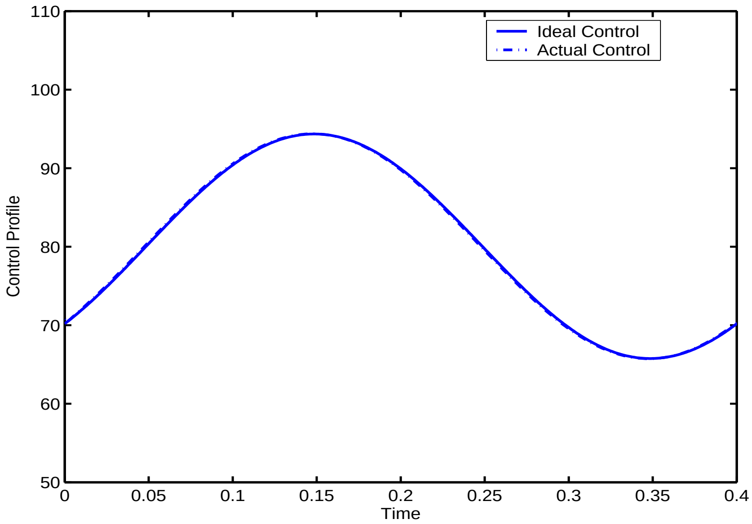

Figure 8.11. Ideal and actual control profiles at 2th period

design for the worst case. It is not needed to use high gain feedback in some particle problems.

The results show that repetitive learning control offers a low feedback gain control, meanwhile achieves the excellent tracking performance. This is owing to its learning functionality as shown in Figure 8.2 and Figure 8.7.

8.7 Conclusion

In this chapter, a new repetitive learning control approach is developed to handle a class of tracking control problems by making use of the repetitive nature of the control problems. The target trajectory can be any smooth periodic orbit of a nonlinear reference model. What can be learned in RLC are either the desired periodic control signals or the lumped uncertainties which may become periodic when the system states converge to the periodic orbit of the reference model.

The repetitive learning control methodology is established with mathematical rigorousness: we first prove the existence of solution by applying the existence theorem of neutral differential difference equation, and using the Lyapunov-Krasovskii functional. Robustifying the repetitive learning control methods with projection and damping has also been exploited in a systematic manner via the Lyapunov-Krasovskii functional approach. As an extension, the integration of RLC and robust adaptive control has also been exploited to address systems with unknown input coefficients and the cascaded systems without strict matching condition. Simulation results exhibited the effectiveness of the new learning control approach.

To recap, the following scenarios were addressed.

1) Nonlinear systems in companion form with unknown but matched nonlinearity which is local Lipschitz continuous, and yielding asymptotic convergence in square integration over one period.

2) Similar scenario like 1) but assuming a known bound on the ideal control profile, yielding uniform asymptotic convergence.

3) Similar scenario like 1) but using a damped learning mechanism, and yielding finite solution trajectory.

4) Similar scenario like 1) but having an unknown input coefficient, leading to a revised learning control law and yielding asymptotic convergence in square integration over one period.

5) Cascaded nonlinear systems with unknown nonlinearities that are local Lipschitz continuous, leading to the integration of robust adaptive and repetitive learning control, and yielding a finite solution trajectory which can be made arbitrarily close to the reference trajectory.

Chapter 9

Multi-Period Repetitive Learning Control with Application to Chaotic Synchronization

9.1 Introduction

Since the chaos synchronization problem was discussed by Pecora and Carroll in 1990 (Pecora and Carroll, 1990), it has received increasing attention. Chaos synchronization has been widely studied in secure communication, chemical reactor and biomedical science. Since chaotic signals could be adopted to transmit information from a master system to a slave system in a secure and robust manner, chaos synchronization has been well studied in communications research (Cuomo *et al.*, 1993), (Chua *et al.*, 1996) and (Dedieu and Ogorzalek, 1997). In (Wu *et al.*, 1996), (Wang and Wang, 1998) and (Zhang *et al.*, 1998), an adaptive method for synchronization of chaotic systems was presented. In (Suykens *et al.*, 1997), a robust nonlinear H_∞ synchronization method was proposed for chaotic Lur'e systems with applications to secure communications. In (Pogromsky, 1998), the problem of controlled synchronization of nonlinear systems was addressed using a passivity-based design method. In (Yu and Song, 2001), an invariant manifold based chaos synchronization approach was proposed. To use only partial states of a chaotic system to synchronize the coupled chaotic systems. In (Song *et al.*, 2002), synchronization to a specific periodic orbit was considered.

It has been shown that many well-known chaotic systems, including Duffing oscillator, R össler system, Chua's circuits, etc., can be transformed into the form of nonlinear dynamical systems with either unknown constant parameters or unknown time-varying factors. Adaptive control methods can well handle chaotic systems with unknown constant parameters (Wang and Ge, 2001*a*) and (Wang and Ge, 2001*b*). On the other hand, the learning control method (Song *et al.*, 2002) has been applied to chaotic systems in the presence of time-varying uncertainties with a uniform periodicity. This chapter considers two new problems in comparison with the previous works (Wang and Ge, 2001*a*), (Wang and Ge, 2001*b*) and (Song *et al.*, 2002). First, the classical adaptive updating law and the periodic learning law are used jointly for systems with both time-varying and time invariant parameters. Generally speaking, the classical adaptive updating law does not work for time varying parameters. The periodic learning control law, on the other hand, does not perform as well as classical adaptive updating law for time invariant parameters due to smoothness problem. Second, the periodic learning law in (Song *et al.*, 2002) only works for a single periodicity, that is, all time varying factors must have the uniform period. In synchronization of two chaotic processes, the master and slave systems may not share a minimum common period, hence we need to address the pseudo-periodicity problem.

To solve the above two problems, it is imperative to develop a new theoretic framework such that the new learning control mechanism can be derived to achieve the global stability and asymptotical synchronization property. We propose a Lyapunov-Krasovskii functional to unify the classical adaptive updating mechanism and the periodic learning mechanism of multiple periods. The asymptotical synchronization is obtained by tuning a chaotic system to follow up a chaotic orbit generated by another chaotic system. It shall be noted that, from point of view of trajectory tracking, the target trajectory now is chaotic, i.e. non-periodic in nature. Hence this chapter extends the previous work (Song *et al.*, 2002) in that a chaotic orbit, instead of a periodic orbit, is considered.

This chapter is organized as follows. Section 9.2 gives the problem formulation. The learning control scheme is presented in Section 9.3. Section 9.4 illustrates a simulation example. The conclusion is given in Section 9.5.

9.2 Problem Formulation

The chaos synchronization problem can often be formulated as for the slave system to follow up the master system. Here the control task is to force the response of the slave system to be synchronized to the chaotic orbit of the master system.

For simplicity, consider the master system Σ_m and slave system Σ_s each with only two unknown parameters, one time varying and one time invariant, as the following

$$\begin{aligned} \Sigma_m \quad \dot{x}_{r,i} &= x_{r,i+1}, \quad i = 1, \cdots n-1, \\ \dot{x}_{r,n} &= \theta_{r_1}\xi_{r_1}(\mathbf{x}_r, t) + \theta_{r_2}(t)\xi_{r_2}(\mathbf{x}_r, t), \end{aligned} \tag{9.1}$$

$$\begin{aligned} \Sigma_s \quad \dot{x}_i &= x_{i+1}, \quad i = 1, \cdots n-1, \\ \dot{x}_n &= -\theta_1\xi_1(\mathbf{x}, t) - \theta_2(t)\xi_2(\mathbf{x}, t) + u(t), \end{aligned} \tag{9.2}$$

where $\mathbf{x}_r = [x_{r,1}, \cdots, x_{r,n}]^T \in \mathcal{R}^n$ and $\mathbf{x} = [x_1, \cdots, x_n]^T \in \mathcal{R}^n$ are the state vectors of the master and slave systems respectively. $\xi_{r_1}(\mathbf{x}_r, t)$, $\xi_{r_2}(\mathbf{x}_r, t)$, $\xi_1(\mathbf{x}, t)$ and

$\xi_2(\mathbf{x}, t)$ are known nonlinear functions which can be locally Lipschitz. θ_{r_1} and θ_1 are unknown constants and $\theta_{r_2}(t),\ \theta_2(t) \in \mathcal{C}[0,\ \infty)$ are unknown continuous periodic function with known periods T_1 and T_2 respectively. The unknown parameters $\theta_1,\ \theta_2,\ \theta_{r_1}(t)$ and $\theta_{r_2}(t)$ should be learned. Note that the negative sign "$-$" in $\dot{x}_n$ can be removed easily by redefining the known functions ξ_1 and ξ_2 with extra negative signs. Adding the negative signs in the slave system is to unify the later derivations. The nonlinear systems (9.1) and (9.2) can be either single-input single-output, or multi-input multi-output, with matched uncertainties of time-invariant or time-varying types.

Note that if there exists a minimum common period T such that for T_1 and T_2, there exist integer numbers m_1 and m_2 satisfying $T = m_1 T_1 = m_2 T_2$, then we can treat the problem with a single-period T. In this chapter, we consider the pseudo-periodic problem in which such a minimum common period T does not exists, for instance $T_1 = \sqrt{2}$ and $T_2 = 2$.

Define the tracking error $e_i(t) = x_{r,i}(t) - x_i(t), i = 1, 2, \cdots, n$ and

$$\sigma(t) = e_n(t) + c_{n-1}e_{n-1}(t) + \cdots + c_1 e_1(t),$$

where $c_i > 0, \quad i = 1, 2, \cdots, n-1$ are coefficients of a Hurwitz polynomial. The synchronization task is to force the slave system Σ_s to track the orbit of the master system by designing an appropriate control input $u(t)$, i.e. let the states of the slave system (9.2) to be asymptotically synchronized with the states of the master system (9.1) as follows

$$\lim_{t\to\infty} \int_{t-T}^{t} \sigma^2(\tau) d\tau = 0. \tag{9.3}$$

In the following we summarize two important properties associated with functionals, which will be used in subsequent derivations with the Lyapunov-Krasovskii functional.

Property 9.1. *Let $\theta(t) \in R$ and $T > 0$ be a finite constant. The upper right-hand derivative of*

$$\int_{t-T}^{t} \theta^2(\tau) d\tau$$

is

$$\theta^2(t) - \theta^2(t-T).$$

Proof. See Appendix A.7. □

Property 9.2. *Let $\theta(t)$, $\hat{\theta}(t)$, $\tilde{\theta}(t)$, $f(t) \in R$, and assume that the following relations hold*

$$\begin{aligned} \theta(t) &= \theta(t-T) \\ \tilde{\theta}(t) &= \theta(t) - \hat{\theta}(t) \\ \hat{\theta}(t) &= \hat{\theta}(t-T) + f(t). \end{aligned} \tag{9.4}$$

Then the upper right-hand derivative of

$$\int_{t-T}^{t} \tilde{\theta}^2(\tau) d\tau$$

is

$$-2\tilde{\theta}(t) f(t) - f^2(t).$$

Proof. See Appendix A.8. □

9.3 Learning Controller Design

The learning control law is

$$\begin{aligned} u(t) &= \beta\sigma(t) + \eta(t) + \hat{\theta}_{r_1}(t)\xi_{r_1}(\mathbf{x}_r, t) + \hat{\theta}_1(t)\xi_1(\mathbf{x}, t) \\ &\quad + \hat{\theta}_{r_2}(t)\xi_{r_2}(\mathbf{x}_r, t) + \hat{\theta}_2(t)\xi_2(\mathbf{x}, t) \end{aligned} \tag{9.5}$$

and the parametric updating law is given as below

$$\begin{cases} \dot{\hat{\theta}}_{r_1}(t) &= \sigma\xi_{r_1}(\mathbf{x}_r, t), \\ \dot{\hat{\theta}}_1(t) &= \sigma\xi_1(\mathbf{x}, t), \\ \hat{\theta}_{r_2}(t) &= \hat{\theta}_{r_2}(t - T_1) + \sigma\xi_{r_2}(\mathbf{x}_r), \\ \hat{\theta}_2(t) &= \hat{\theta}_2(t - T_2) + \sigma\xi_2(\mathbf{x}), \end{cases} \tag{9.6}$$

where $\eta(t) = c_{n-1}e_n(t) + \cdots + c_1e_2(t)$. The parametric updating law (9.6) is a part of the control law, in the sequel the controller is dynamic in nature. Without the loss of generality, assume $T_2 \geq T_1$. At the initial period $t \in [0, T_1]$, $\hat{\theta}_{r_2}(t) = \sigma\xi_{r_2}(\mathbf{x}_r)$. Similarly at the initial period $t \in [0, T_2]$, $\hat{\theta}_2(t) = \sigma\xi_2(\mathbf{x})$, For notational convenience, we will omit the argument t for all variables where no confusion arises, and denote $\xi_{r_i}(\mathbf{x}_r, t)$ and $\xi_i(\mathbf{x}, t)$ by ξ_{r_i} and ξ_i, respectively for $i = 1, 2$. It should be noted that the parametric updating law is actually a mixture with the classical parametric adaptation and periodic learning mechanisms.

Substituting the control law (9.5) with the mixed parametric learning law (9.6) into the dynamics (9.2) yields the error dynamics

$$\begin{aligned} \dot{e}_i &= \dot{x}_{r,i} - \dot{x}_i = e_{i+1}, \quad i = 1, 2, \cdots, n-1. \\ \dot{e}_n &= \dot{x}_{r,n} - \dot{x}_n \\ &= \theta_{r_1}\xi_{r_1} + \theta_{r_2}(t)\xi_{r_2} + \theta_1\xi_1 + \theta_2(t)\xi_2 \\ &\quad -[\beta\sigma + \eta + \hat{\theta}_{r_1}(t)\xi_{r_1} + \hat{\theta}_{r_2}(t)\xi_{r_2} + \hat{\theta}_1(t)\xi_1 + \hat{\theta}_2(t)\xi_2] \\ &= -\beta\sigma + \phi_{r_1}\xi_{r_1} + \phi_{r_2}\xi_{r_2} + \phi_1\xi_1 + \phi_2\xi_2 - \eta \end{aligned} \tag{9.7}$$

where

$$\begin{aligned} \phi_i &= \theta_i - \hat{\theta}_i, \\ \phi_{r_i} &= \theta_{r_i} - \hat{\theta}_{r_i}. \end{aligned}$$

for $i = 1, 2$. Accordingly we can derive

$$\begin{aligned} \dot{\sigma} &= \dot{e}_n(t) + c_{n-1}e_n(t) + \cdots + c_1e_2(t) \\ &= -\beta\sigma + \phi_{r_1}\xi_{r_1} + \phi_1\xi_1 + \phi_{r_2}\xi_{r_2} + \phi_2\xi_2. \end{aligned} \tag{9.8}$$

To facilitate the convergence analysis, define the following Lyapunov-Krasovskii functional

$$V(t,\sigma,\phi_1,\phi_2,\phi_{r_1},\phi_{r_2}) = \begin{cases} \frac{1}{2}\sigma^2(t)+\frac{1}{2}\phi_{r_1}^2(t)+\frac{1}{2}\phi_1^2(t)+\frac{1}{2}\int_{t-T_1}^{t}\phi_{r_2}^2(\tau)d\tau+\frac{1}{2}\int_{t-T_2}^{t}\phi_2^2(\tau)d\tau & t\in[T_2,\infty) \\ \frac{1}{2}\sigma^2(t)+\frac{1}{2}\phi_{r_1}^2(t)+\frac{1}{2}\phi_1^2(t)+\frac{1}{2}\int_{t-T_1}^{t}\phi_{r_2}^2(\tau)d\tau+\frac{1}{2}\int_{0}^{t}\phi_2^2(\tau)d\tau & t\in[T_1,T_2) \\ \frac{1}{2}\sigma^2(t)+\frac{1}{2}\phi_{r_1}^2(t)+\frac{1}{2}\phi_1^2(t)+\frac{1}{2}\int_{0}^{t}\phi_{r_2}^2(\tau)d\tau+\frac{1}{2}\int_{0}^{t}\phi_2^2(\tau)d\tau & t\in[0,T_1) \end{cases}$$

The convergence property of the proposed adaptive control method is summarized in the following theorem.

Theorem 9.1. *The control law (9.5) with the parametric updating law parameter law (9.6) warrants the asymptotical convergence*

$$\lim_{t\to\infty}\int_{t-T_2}^{t}\sigma^2(\tau)d\tau = 0.$$

Proof. The proof consists three parts. Part I proves the finiteness of V in $[0,\ T_2)$. Part II proves the negativeness of V in $[T_2,\ \infty)$. Part III derives the asymptotical convergence of the tracking error $\sigma(t)$.

Part I: Finiteness of V in $[0,\ T_2)$

Let us first derive the upper right hand derivative of V for $t \in [0, T_1)$, which is

$$\dot{V} \;=\; \sigma\dot{\sigma}+\phi_{r_1}\dot{\phi}_{r_1}+\phi_1\dot{\phi}_1+\frac{1}{2}\phi_{r_2}^2(t)+\frac{1}{2}\phi_2^2(t) \tag{9.9}$$

Look into the first term on the right hand side of $\dot{V}$. From (9.8), we obtain

$$\sigma\dot{\sigma} = -\beta\sigma^2+\phi_{r_1}(t)\xi_{r_1}\sigma+\phi_1(t)\xi_1\sigma+\phi_{r_2}(t)\xi_{r_2}\sigma+\phi_2(t)\xi_2\sigma. \tag{9.10}$$

Using the parametric updating law (9.6), we have

$$\phi_{r_1}\dot{\phi}_{r_1} = -\phi_{r_1}\xi_{r_1}\sigma, \tag{9.11}$$

and

$$\phi_1\dot{\phi}_1 = -\phi_1\xi_1\sigma. \tag{9.12}$$

For $t \in [0,\ T_1)$, $\hat{\theta}_{r_2} = \sigma\xi_{r_2}$ and $\hat{\theta}_2 = \sigma\xi_2$, therefore

$$\begin{aligned}\phi_{r_2}^2(t) &= (\theta_{r_2}(t) - \hat{\theta}_{r_2}(t))^2 \\ &= \theta_{r_2}^2(t) - 2\hat{\theta}_{r_2}(t)\phi_{r_2}(t) - \hat{\theta}_{r_2}^2(t) \\ &\leq \theta_{r_2}^2(t) - 2\phi_{r_2}(t)\xi_{r_2}\sigma,\end{aligned}$$

and similarly

$$\phi_2^2(t) \leq \theta_2^2(t) - 2\phi_2(t)\xi_2\sigma.$$

In the sequel, the upper right hand derivation of V for $t \in [0,\ T_1)$ is

$$\dot{V} \leq -\beta\sigma^2 + \frac{1}{2}\theta_{r_2}^2(t) + \frac{1}{2}\theta_2^2(t).$$

Note that $\theta_{r_2}(t)$ and $\theta_2(t)$ are periodic, thus are bounded. The finiteness of $\dot{V}$ warrants the finiteness of V in a finite time interval $[0,\ T_1)$.

For $t \in [T_1,\ T_2)$, the upper right hand derivative of V according to Property 9.1 is

$$\dot{V} = \sigma\dot{\sigma} + \phi_{r_1}\dot{\phi}_{r_1} + \phi_1\dot{\phi}_1 + \frac{1}{2}(\phi_{r_2}^2(t) - \phi_{r_2}^2(t - T_1)) + \frac{1}{2}\phi_2^2(t), \qquad (9.13)$$

where $\sigma\dot{\sigma}$, $\phi_{r_1}\dot{\phi}_{r_1}$ and $\phi_1\dot{\phi}_1$ can be achieved from (9.10), (9.11) and (9.12). According to Property 9.2 and the parameter updating law (9.6), we have

$$\phi_{r_2}^2(t) - \phi_{r_2}^2(t - T_1) = -2\phi_{r_2}(t)\xi_{r_2}\sigma - \xi_{r_2}^2\sigma^2.$$

For $t \in [T_1,\ T_2)$, we still have $\hat{\theta}_2 = \sigma\xi_2$, thus

$$\phi_2^2(t) \leq \theta_2^2(t) - 2\phi_2(t)\xi_2\sigma.$$

Therefore, the upper right hand derivation of V for $t \in [T_1,\ T_2)$ is

$$\dot{V} \leq -\beta\sigma^2 - \frac{1}{2}\xi_{r_2}^2\sigma^2 + \frac{1}{2}\theta_2^2(t)$$

Obviously $\dot{V}$ is finite for $t \in [T_1,\ T_2)$ because of the finiteness of the periodic function $\theta_2(t)$. This implies that V is bounded for t in a finite time interval $[T_1,\ T_2)$.

Part II: Negativeness of V in $[T_2, \infty)$

The upper right hand derivative of V, according to Property 9.1 for $t \in [T_2, \infty)$, should be

$$\begin{aligned}\dot{V} &= \sigma\dot{\sigma} + \phi_{r_1}\dot{\phi}_{r_1} + \phi_1\dot{\phi}_1 \\ &\quad + \frac{1}{2}(\phi_{r_2}^2(t) - \phi_{r_2}^2(t - T_1)) + \frac{1}{2}(\phi_2^2(t) - \phi_2^2(t - T_2)). \qquad (9.14)\end{aligned}$$

Considering the terms on the right hand side of $\dot{V}$ in (9.14), $\sigma\dot{\sigma}$, $\phi_{r_1}\dot{\phi}_{r_1}$ and $\phi_1\dot{\phi}_1$ are the same as (9.10), (9.11) and (9.12). According to Property 9.2 and the parameter updating law (9.6), we can further derive the following relationship

$$\phi_{r_2}^2(t) - \phi_{r_2}^2(t - T_1) = -2\phi_{r_2}(t)\xi_{r_2}\sigma - \xi_{r_2}^2\sigma^2,$$

and

$$\phi_2^2(t) - \phi_2^2(t - T_2) = -2\phi_2(t)\xi_2\sigma - \xi_2^2\sigma^2.$$

Therefore, the upper right hand derivation of V is

$$\begin{aligned}\dot{V} &= -\beta\sigma^2 - \frac{1}{2}\xi_{r_2}^2\sigma^2 - \frac{1}{2}\xi_2^2\sigma^2 \\ &\leq -\beta\sigma^2. \qquad (9.15)\end{aligned}$$

Part III: Asymptotical Convergence

Now let us derive the convergence property

$$\lim_{t\to\infty}\int_{t-T_2}^{t}\sigma^2(\tau)d\tau = 0$$

using the fact (9.15) that $\dot{V}$ for $t \in [T_2, \infty)$ is negative semi-definiteness. Suppose that

$$\lim_{t\to\infty}\int_{t-T_2}^{t}\sigma^2(\tau)d\tau \neq 0.$$

Then there exist an $\epsilon > 0$, a $t_0 \geq T_2$ and a sequence $t_i \to \infty$ with $i = 1, 2, \cdots$ and $t_{i+1} \geq t_i + T_2$ such that $\int_{t_i - T_2}^{t_i}\sigma^2(\tau)d\tau > \epsilon$ when $t_i > t_0$. Hence from (9.15), we obtain for $t > T_2$

$$\begin{aligned}\lim_{i\to\infty} V(t,\sigma,\phi_{r_1},\phi_1,\phi_{r_2},\phi_2) &\leq V(T_2,\sigma(T_2),\phi_{r_1}(T_2),\phi_1(T_2),\phi_{r_2}(T_2),\phi_2(T_2)) \\ &\quad - \lim_{i\to\infty}\sum_{j=1}^{i}\int_{t_j-T_2}^{t_j}\beta\sigma^2(\tau)d\tau.\end{aligned}$$

Since $V(T_2,\sigma(T_2),\phi_{r_1}(T_2),\phi_1(T_2),\phi_{r_2}(T_2),\phi_2(T_2))$ is finite, the above relation implies

$$\lim_{t\to\infty} V(t,\sigma,\phi_{r_1},\phi_{r_2},\phi_1,\phi_2) \to -\infty,$$

a contradiction to the positiveness property of $\lim_{t\to\infty} V(t,\sigma,\phi_{r_1},\phi_1,\phi_{r_2},\phi_2)$.

This completes the proof. □

Remark 9.1. *The above result can be extended straightforward to the master system*

$$\begin{aligned}\dot{x}_{r,i} &= x_{r,i+1}, \quad i = 1,\cdots n-1, \\ \dot{x}_{r,n} &= \boldsymbol{\theta}_{r_1}\boldsymbol{\xi}_{r_2}(\mathbf{x}_r,t) + \boldsymbol{\theta}_{r_2}(t)\boldsymbol{\xi}_{r_2}(\mathbf{x}_r,t), \end{aligned} \tag{9.16}$$

and the slave system

$$\begin{aligned}\dot{x}_i &= x_{i+1}, \quad i = 1,\cdots n-1, \\ \dot{x}_n &= \boldsymbol{\theta}_1(t)\boldsymbol{\xi}_1(\mathbf{x},t) + \boldsymbol{\theta}_2(t)\boldsymbol{\xi}_2(\mathbf{x},t) + u(t), \end{aligned} \tag{9.17}$$

where $\boldsymbol{\theta}_{r_1},\ \boldsymbol{\theta}_1 \in \mathcal{R}^m$, $\boldsymbol{\theta}_{r_2},\ \boldsymbol{\theta}_2 \in \mathcal{C}^m[0,\infty)$ are vector valued functions, and

$$\begin{aligned}\boldsymbol{\xi}_{r_i}(\mathbf{x}_r,t) &= [\xi_{r_i,1}(\mathbf{x}_r,t),\xi_{r_i,2}(\mathbf{x}_r,t),\cdots,\xi_{r_i,m}(\mathbf{x}_r,t)]^T \\ \boldsymbol{\xi}_i(\mathbf{x},t) &= [\xi_{i,1}(\mathbf{x},t),\xi_{i,2}(\mathbf{x},t),\cdots,\xi_{i,m}(\mathbf{x},t)]^T,\end{aligned}$$

for $i = 1,2$. Accordingly we can replace $\hat{\theta}_{r_i}(t)$, $\hat{\theta}_i(t)$ by $\hat{\boldsymbol{\theta}}_{r_i}(t)$, $\hat{\boldsymbol{\theta}}_i(t)$ and $\xi_{r_i}(\mathbf{x}_r,t)$, $\xi_i(\mathbf{x},t)$ by $\boldsymbol{\xi}_{r_i}(\mathbf{x}_r,t)$, $\boldsymbol{\xi}_i(\mathbf{x},t)$ in the learning mechanism, and replace ϕ_i^2 and $\phi_{r_i}^2$ in Lyapunov function by $\boldsymbol{\phi}_i^T\boldsymbol{\phi}_i$ and $\boldsymbol{\phi}_{r_i}^T\boldsymbol{\phi}_{r_i}$, for $i = 1,2$.

9.4 Illustrative Example

Consider the master system to be the Duffing system

$$\begin{aligned}\dot{x}_{r,1} &= x_{r,2},\\ \dot{x}_{r,2} &= \theta_{r_1} x_{r,1} + \theta_{r_2} x_{r,1} - x_{r,1}^3 + \theta_{r_3}(t).\end{aligned} \qquad (9.18)$$

With $\theta_{r_1} = 1.1$, $\theta_{r_2} = -0.4$ and $\theta_{r_3}(t) = 1.8cos(1.8t)$, the system generates a chaotic orbit seen in Figure 9.1.

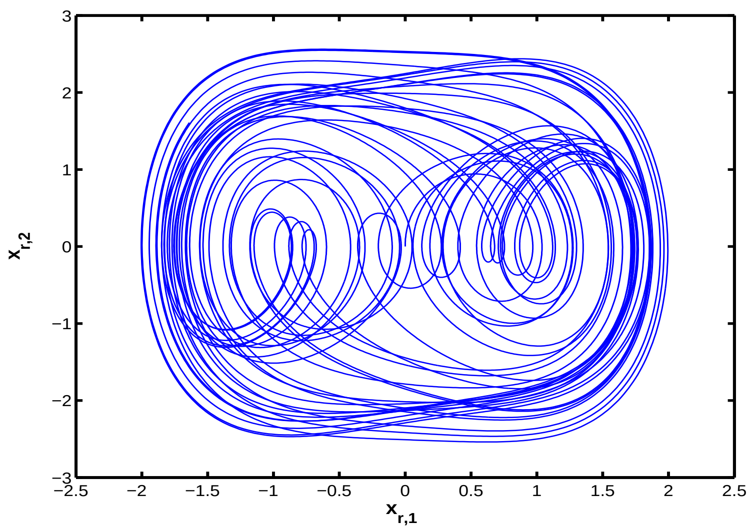

Figure 9.1. Chaotic Orbit of the Duffing System ($x_{r,1} = 0, x_{r,2} = 0.$)

The slave system is

$$\begin{aligned}\dot{x}_1 &= x_2,\\ \dot{x}_2 &= \theta_1 x_1 + \theta_2 x_2 - x_1^3 + \theta_3(t) + u(t),\end{aligned} \qquad (9.19)$$

where $\theta_1 = 1$, $\theta_2 = -0.25$ and $\theta_3(t) = 0.3\cos t$. In the example, $T_1 = 2\pi/1.8$ and $T_2 = 2\pi$. We treat the problem as with different periods, though a unified period $T = 3.6\pi$ exists. The learning process will be delayed by using a larger period.

Without any control, i.e., $u = 0$, the slave system also generates a chaotic orbit shown in Figure 9.2.

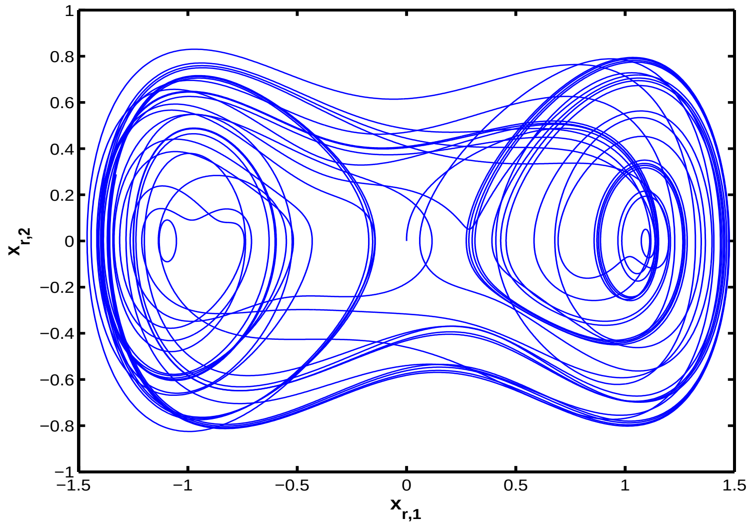

Figure 9.2. Chaotic Orbit of the slave System without controller ($x_1 = 0, x_2 = 0.$)

Figure 9.1 and Figure 9.2 show that the two systems have the different chaotic orbit. Our objective is to design a controller $u(t)$ such that the chaotic orbit of the slave system will be synchronized to the master system. Based on the learning control design given in Section 3, the simulation results are given in the following. Figure 9.3 and Figure 9.4 show the states of slave system after 10−th periods and 50−th periods respectively.

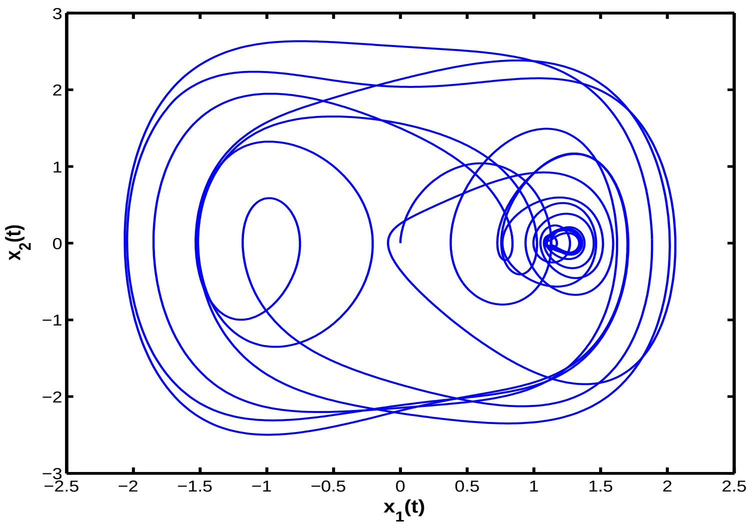

Figure 9.3. Chaotic Orbit of the slave System after 10−th period.

It can be seen that the orbit of Figure 9.4 is almost the same as Figure 9.1. Figure

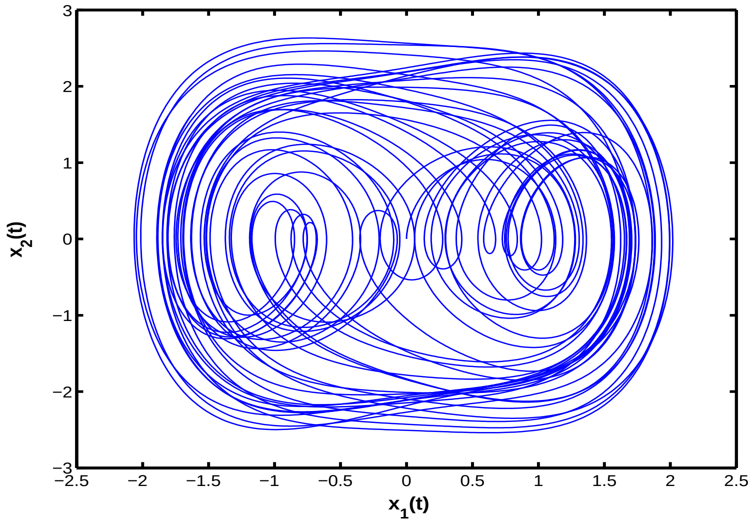

Figure 9.4. Chaotic Orbit of the slave System after 50−th period.

9.5 displays the tracking error σ. In the figure, $|\sigma_i|_{sup}$ is used to record the maximum absolute tracking error during the i−th period.

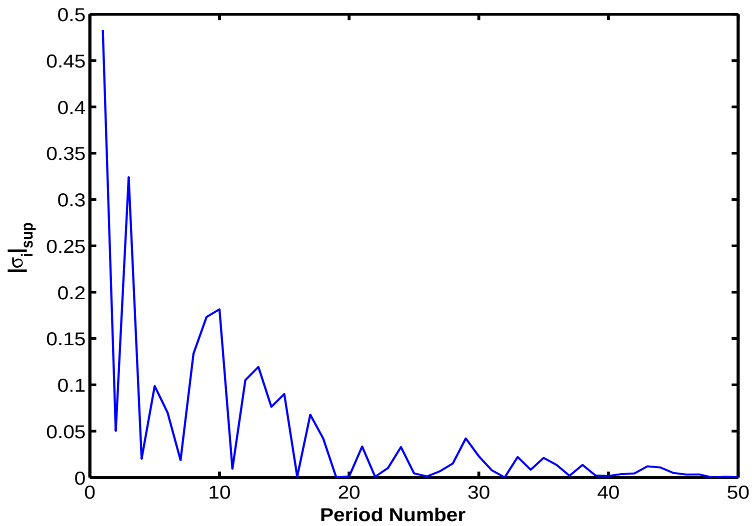

Figure 9.5. Tracking Error $\sigma(t)$ Convergence

Finally, to show the advantage of the mixed parameter updating law, the periodic updating law is applied to the time invariant parameters θ_{r_1} and θ_1. The tracking error is shown in the Figure 9.6. Comparing with the preceding results, the effectiveness of the new learning control method in the synchronization is immediately obvious.

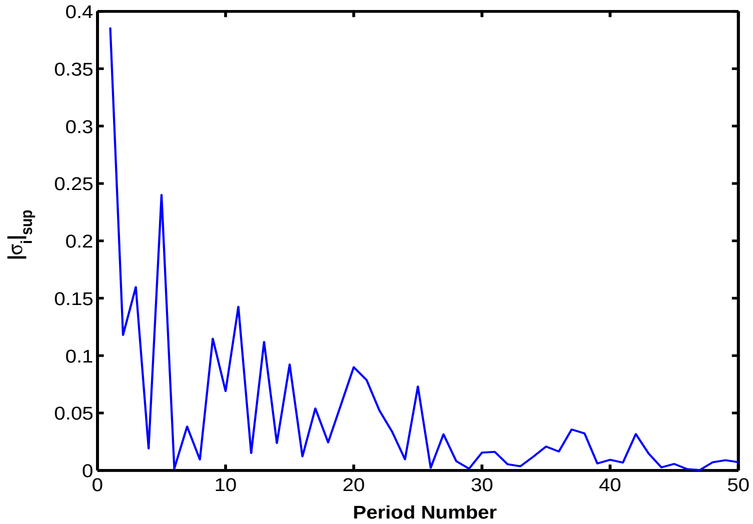

Figure 9.6. Tracking Error $\sigma(t)$ for the periodic updating law applied to the time invariant parameters θ_{r_1} and θ_1

9.5 Conclusion

A learning control approach for synchronization of two uncertain chaotic systems was presented. Global stability and asymptotic synchronization have been achieved for chaotic systems with both time-varying and time invariant parametric uncertainties. The validity of the new approach is confirmed through theoretical analysis and numerical simulations.

Chapter 10

Conclusions and Future Research

10.1 Conclusions

In this thesis, several learning control approaches are presented for linear and nonlinear dynamic systems. The contribution of this research work is to investigate and analyze learning control, disclose the inherent nature of learning control, and therefore facilitate the design of learning control.

The objective of direct learning is to generate the desired control profile for a newly switched system without any feedback, even if the system may have uncertainties. A DLC scheme is achieved by exploring the inherent relationship between any two systems before and after a switch. In Chapter 2, a DLC approach for a class of switched systems has been proposed. The approach is applicable to a class of linear time varying, uncertain, and switched systems, when the trajectory tracking control problem is concerned. Furthermore, singularity problem and trajectory switch problem are also considered.

After the formalization by Arimoto, iterative learning control has attracted increased interesting for systems with repetitive operation. However, the early researches have designed a iterative learning control system in the presence of input nonsingularity. In Chapter 3, two kinds of ILC approaches have been presented by adding a forgetting factor and adopting a time varying learning gain to deal with input singularities problem. The proposed ILC approaches ensure a convergent control input sequence approaching to a unique fixed point based on Banach fixed point theorem. In the presence of the first type of singularities, the fixed point guarantees that the system output enters and remains uniformly in a designated neighborhood of the target trajectory. While in the presence of the second type of singularities, the tracking error is bounded by a class $\mathcal{K}$ function of the designated neighborhood.

In Chapter 4, the attention has been concentrated on exploring the possibility of designing an ILC scheme for systems without a priori knowledge of the control direction. By incorporating a Nussbaum-type function, a new learning control mechanism has been constructed with both differential and difference updating laws. The new learning control mechanism can warrant a L_T^2 convergence of the tracking error sequence along the iteration axis, in the presence of time-varying parametric uncertainties and local Lipschitz nonlinearities.

A constructive function approximation approach has been proposed for adaptive learning control which handles finite interval tracking problems in Chapter 5. Unlike the well established adaptive neural control which uses a fixed neural network structure as a complete system, in the method the function approximation network consists of a set of bases and the number of bases can be increased when learning repeats. The nature of basis allows the continuously adaptive tuning or learning of parameters when the network undergoes a structure change, consequently offers the flexibility in tuning the network structure. The expansibility of the basis ensures the function approximation accuracy, and removes the processes in pre-setting the network size. Two classes of system unknown nonlinear functions, either in $\mathcal{L}^2(R)$

or a known upperbound, are taken into consideration. With the help of Lyapunov method, the existence of solution and the convergence property of the proposed adaptive learning control system, are analyzed rigorously.

Initial conditions, or initial resetting conditions, play a fundamental role in all kinds of iterative learning control methods. In Chapter 6, five different initial conditions have been studied to disclose the inherent relationship between each initial condition and corresponding learning convergence (or boundedness) property. The ILC approach under consideration is based on Lyapunov theory, which is suitable for plants with time varying parametric uncertainties and local Lipschitz nonlinearities.

A new RLC approach has been developed for systems with unknown periodic parameters in Chapter 7. With mathematical rigorousness the existence of solution and learning convergence are proved. Robustifying the nonlinear learning control with projection and forgetting factor has also been exploited in a systematic manner via the Lyapunov-Krasovskii functional approach.

In Chapter 8, an RLC approach has been proposed to deal with periodic tracking tasks for nonlinear dynamical systems with non-parametric uncertainties. Three fundamental issues are addressed associated with the new learning control methodology: the existence of the solution, learning convergence property and robustification, which are indispensable for the new learning control framework. Applying the existence theorem of the differential difference equation of neutral type, and using Lyapunov-Krasovskii functional, the existence of solution and the learning convergence can be proven rigorously. To enhance the robustness of the repetitive learning control, two kinds of robustification methods are developed with projection and damping respectively to ensure the boundedness of the learning signals. A further extension of RLC to more general nonlinear systems with unmatched uncertainties has been also exploited.

As an application, a learning control approach for synchronization of two uncertain chaotic systems has been presented in Chapter 9. Global stability and asymptotic synchronization have been achieved for chaotic systems with both time-varying and time invariant parametric uncertainties.

10.2 Suggestions for the Future Research

Past research activities have laid a foundation for the future work. Based on the prior research, the following problems deserve further consideration and investigation.

1. From Chapter 3 and Chapter 4, it is known that contraction mapping method is a systematic way of analyzing learning convergence based on the global Lipschitz condition and composite energy function based ILC convergence analysis is widely applied to nonlinear systems. It is worth to note that the contraction mapping based learning enjoys a geometric convergence speed, which is far better than the asymptotic convergence of energy function based learning. Can the two methods be combined together to improve the convergence effect? For instance, the simplest idea is to adopt energy method for a nonlinear system first, then switch to contraction mapping method when the tracking error enters or lies in a neighborhood. However it is not clear how to describe and estimate the range of the neighborhood, and how to deal with the relative degree problem.

2. In Chapter 5, Chapter 7 and Chapter 8, the tracking problem for a class of nonlinear dynamic systems with either parametric uncertainty or non-parametric uncertainty have been studied based on Lyapunov-Krasovskii

functional method and constructive function approximation. Are there any other analytic approaches better solve the problem?

3. In contraction mapping method, can the transient response of the system in time domain be discussed?

4. Can CEF method be extended to deal with non-affine dynamic systems?

5. The convergence speed of contraction mapping method based learning has been calculated in the previous works, then can the convergence speed of Lyapunov-Krasovskii functional method based learning be estimated?

6. For discrete-time systems, there is a lot of work done in the field of contraction mapping based learning. What is the discrete-time version of Lyapunov-Krasovskii functional method based learning?

7. In the previous Chapters, Lyapunov-Krasovskii functional method based learning requires the states be physically measurable. To solve the output tracking without using the system state information, learning control needs to combine with state estimation. In such case, non-minimum phase will be an obstacle.

8. In fact, learning control that study at present is based on the numerical approximation, and are not able to give an analytic expression, even if the learning converges. Whether an analytic function can be found iteratively to yield an appropriate control is a highly challenging problem.

9. Can the learning control be merged with other types of learning methods, such as neural learning, statistical learning, machine learning, etc, to come up with a new paradigm of intelligent control system theory?

There are still many open problems in the area of learning control, waiting for us to explore and solve.

Bibliography

Ahn, H.S., C.H. Choi and K.B. Kim (1993). Iterative learning control for a class of nonlinear systems. *Automatica* **29**(6), 1575–1578.

Apostol, T. M. (1957). *Mathematical analysis*. MA: Addison-Wesley.

Arimoto, S., S. Kawamura and F. Miyazaki (1984*a*). Bettering operation of robots by learning. *J. of Robotic Systems* **1**(2), 123–140.

Arimoto, S., S. Kawamura and F. Miyazaki (1984*b*). Iterative learning control for robot systems. In: *Proceedings of IECON*. Tokyo, Japan. pp. 127–134.

Arimoto, S., T. Naniwa and H. Suzuki (1991). Selective learning with a forgetting factor for robotic motion control. In: *Proc. of the 1991 IEEE Int. Conf. on Robotics and Automation*. Vol. 9. Sacramento, CA, USA. pp. 728–733.

Branicky, M. S. (1998). Multiple lyapunov functions and other analysis tools for switched and hybrid systems. *IEEE Transactions on Automatic Control* **43**, 475–482.

Brogliato, B. and R. Lozano (1992). Adaptive control of a simple nonlinear system without a priori information on the plant parameters. *IEEE Trans. Automat. Contr.* **37**, 30–37.

Brogliato, B. and R. Lozano (1994). Adaptive control of first-order nonlinear systems with reduced knowledge of the plant parameters. *IEEE Trans. Automat. Contr.* **39**, 1764–1768.

Cao, W. J. and J.-X. Xu (2001). Robust and almost perfect periodic tracking of nonlinear systems using repetitive vsc. In: *In Proceedings of IEEE 2001 American Control Conference*. Arlington, VA, USA. pp. 3830–3835.

Chen, H. D. and P. Jiang (2002). Adaptive iterative feedback control for nonlinear system with unknown high-frequence gain. In: *Proceedings of the 4th World Congress on Intelligent Control and Automation*. Shanghai, P.R. China. pp. 847–851.

Chen, Y., C. Wen, Z. Gong and M. Sun (1999). An iterative learning controller with initial state learning. *IEEE Transactions on Automatic Control* **44**(2), 371–376.

Chen, Y. Q. and C. Y. Wen (1999). *Iterative Learning Control – convergence, robustness and applications*. Vol. 248 of *Lecture Notes in Control and Information Science*. Springer-Verlag. London, UK.

Chen, Yangquan, Zhiming Gong and Changyun Wen (1998). Analysis of a high-order iterative learning control algorithm for uncertain nonlinear systems with state delays. *Automatica* **34**(3), 345–353.

Chien, C. J. (1996). A discrete iterative learning control of nonlinear time-varying systems. In: *Proceedings of the 35th IEEE Conference on Decision and Control*. Vol. 3. Kobe, Japan. pp. 3056–3061.

Chien, C. J. (1998). On the iterative learning control of sampled-date systems. In: *Iterative Learning Control – Analysis, Design, Integration and Applications*

(Z. Bien and J. X. Xu, Eds.). pp. 71–82. Kluwer Academic Press. Boston, USA.

Chien, Chiang-Ju and Chia-Yu Yao (2004). Iterative learning of model reference adaptive controller for uncertain nonlinear systems with only output measurement. *Automatica* **40**(5), 855–864.

Chua, L. O., T. Yang, G. Q. Zhong and C. W. Wu (1996). Adaptive synchronization of chua's oscillators. *International Journal of Bifurcation and Chaos* **6**, 189–201.

Corless, M. and G. Leitmann (1981). Continuous state feedback guaranteeing uniform ultimate boundedness for uncertain dynamic systems. *IEEE Trans. on Aut. Contr.* **26**(5), 1139–1144.

Cruz, M. A. and J. K. Hale (1970). Existence, uniqueness and continuous dependence for hereditary systems. *Annali Mat. pura appl.* **85**(4), 63–82.

Cuomo, K. M., A. V. Oppenheim and S. H. Strogatz (1993). Synchronization of lorenz-based chaotic circuits with applications to communications. *IEEE Trans. Circ. Syst* **40**, 626–633.

Daubechies, I. (1988). Orthonormal bases of compactly supported wavelets. *Comm. Pure and Appl. Math.* **41**, 909–996.

Dedieu, H. and M. J. Ogorzalek (1997). Identifiability and identification of chaotic systems based on adaptive synchronization. *IEEE Trans. Circ. Syst* **44**, 948–962.

Dixon, W. E., E. Zergeroglu, D. M. Dawson and B. T. Costic (2003). Repetitive learning control: A lyapunov-based approach. *IEEE Trans. Syst., Man, Cybern. B* **32**(8), 538–545.

Driver, R. (1965). Existence and continuous dependence of solutions of a neutral functional differential equation. *Arch. Rat. Mech. Aun* **19**, 147–166.

EL'SGOL'TS, L. E. (1964). Qualitative methods in mathematical analysis. *Amer. Math. Soc. Translations of Mathematical Monographs.*

Ezzine, J. and A. H. Haddad (1989). Controllability and observability of hybrid systems. *International Journal of Control* **49**, 2045–2055.

French, J. A. and M. Q. Phan (2000). Linear quadratic optimal learning control (LQL). **73**(10), 832–839.

French, M. and E. Rogers (2000*a*). Non-linear iterative learning by an adaptive lyapunov technique. *Int. J. Control* **73**(10), 840–850.

French, M. and E. Rogers (2000*b*). Non-linear iterative learning by an adaptive lyapunov technique. *International Journal of Control* **73**(10), 840–850.

Fu, Pengcheng and John P. Barford (1992). Simulation of an iterative learning control system for fed-batch cell culture processes. *Cytotechnology* **10**(1), 53–62.

Funahashi, K. (1989). On the approximate realization of continuous mapping by neural networks. *Neural Networks* **2**, 183–192.

Ge, S. S. and C. Wang (2002). Direct adaptive NN control of a class of nonlinear systems. *IEEE Trans. Neural Networks* **13**(1), 214–221.

Ghash, J. and B. Paden (2002). A pseudoinverse-based iterative learning control. *IEEE Trans. of Automatic Control* **47**(5), 831–837.

Gupta, M. M. and D. H. Rao (1994). *Neuro-control systems: Theory and applications.* IEEE Neural Networks Council. New York, NY.

Hale, J. K. and M. A. Pedro (1977). Stability in neural equations. *Nonlinear analysis, theory, methods & applications* **1**(2), 161–173.

Ham, C., Z. Qu and J. Kaloust (2001). Nonlinear learning control for a class of nonlinear systems. *Automatica* **37**(3), 419–428.

Hara, S., Y. Yamamoto, T. Omata and M. Nakano (1988). Repetitive control system: a new type servo system for periodic exogenous signals. *IEEE Trans. of Automatic Control* **33**(7), 659–668.

Heinzinger, G., D. Fenwick, B. Paden and F. Miyazaki (1992). Stability of learning control with disturbances and uncertain initial conditions. *IEEE Trans. of Automatic Control* **37**(1), 110–114.

Hornik, K., M. Stinchcombe and H. White (1989). Multilayer feedforward networks are universal approximators. *Neural Networks* **2**, 359–366.

Huang, Sunan, K. K. Tan and T. H. Lee (2003). Further results on adaptive control for a class of nonlinear systems using neural network. *IEEE Trans. Neural Networks* **14**(3), 719–722.

Hunt, K. J., D. Sbarbaro, R. Zbikowski and P. J. Gawthrop (1992). Neural networks for control system-a survey. *Automatica* **28**(6), 1083–1112.

Ioannou, P. A. and J. Sun (1996). *Robust adaptive control*. Englewood Cliffs. New Jersey: PH.

Jang, T. J., C. H. Choi and H. S. Ahn (1995). Iterative learning control in feedback systems. *Automatica* **31**(2), 243–245.

Ji, Y. and H. J. Chizeck (1988). Controllability, observability and discrete-time markovian jump linear quadratic control. *International Journal of Control* **48**(2), 481–498.

Jiang, P. and R. Unbehauen (2002). Robot visual servoing with iterative learning control. *IEEE Transactions on Systems, Man, and Cybernetics–Part A* **32**(2), 281–287.

Kaloust, J. and Z. Qu (1995). Continuous robust control design for nonlinear uncertain systems without a priori knowledge of control direction. *IEEE Trans. Automat. Contr.* **40**(2), 276–282.

Kelly, S. E., M. A. Kon and L. A. Raphael (1994). Local convergence for wavelet expansions. *J. Funct. Anal* **126**, 102–138.

Kim, Y. H. and I. J. Ha (1999). A learning approach to precision speed control of servomotors and its application to vcr. *IEEE Trans. Contr. Syst. Technol* **7**, 466–477.

Kuc, T. Y., J. S. Lee and K. Nam (1992). An iterative learning control theory for a class of nonlinear dynamic systems. *Automatica* **28**(6), 1215–1221.

Kuc, T. Y., Kwanghee Nam and J. S. Lee (1991). An iterative learning control of robot manipulators. *IEEE Transactions on Robotics and Automation* **7**(1), 835–842.

Lebedev, L. P., I. I. Vorovich and G. M. L. Gladwell (1994). *Functional Analysis.* Kluwer Academic Publishers. Norwell, MA 02061, U. S. A.

Lee, H. S. and Z. Bien (1997). A note on convergence property of iterative learning controller with respect to sup norm. *Automatica* **33**(8), 1591–1593.

Lee, K. H. and Z. Bien (1991). Initial condition problem of learning control. *IEE Proceedings, Part–D, Control Theory and Applications* **138**(6), 525–528.

Lee, K.S. and J.H. Lee (2000). Convergence of constrained model-based predictive control for batch processes. *IEEE Transactions on Automatic Control* **45**(10), 1928–1932.

Lee, Kwang Soon and Jay H. Lee (1997). Constrained model–based predictive control combined with iterative learning for batch or repetitive processes. In: *Proceedings of the 2nd Asian Control Conference*. Vol. 93. Seoul, Korea. pp. 85–101.

Levin, A. U. and K. S. Narendra (1996). Control of nonlinear dynamical systems using neural networks-part ii: Observability, identification, and control. *IEEE Trans. Neural Networks* **7**(1), 30–42.

Liberzon, D. and A. S. Morse (1999). Basic problems in stability and design of switched systems. *IEEE Contr. Syst. Mag.* **19**, 59–70.

Longman, R. W. (1998). Designing iterative learning and repetitive controllers. In: *Iterative Learning Control – Analysis, Design, Integration and Applications* (Z. Bien and J. X. Xu, Eds.). pp. 107–146. Kluwer Academic Press. Boston, USA.

Longman, Richard W. (2000). Iterative learning control and repetitive control for engineering practice. *International Journal of Control* **73**(10), 930–954.

Loparo, K. A., J. T. Aslanis and O. IIajek (1987). Analysis of switching linear systems in the plane, part 2, global behavior of trajectories, controllability and attainability. *J. of Optim. Theory Appl.* **52**(3), 395–427.

Lucibello, P. (1996). On trajectory tracking learning of flexible joint arms. *International Journal of Robotics and Automation* **11**(4), 190–194.

Mallat, S. G. (1989). Multiresolution approximations and wavelet orthonormal bases of $L^2(R)$. *Transactions of the American Mathematical Society* **315**(1), 69–87.

Messner, W. and M. Bodson (1995). Design of adaptive feedforward algorithms using internal model equivalence. *Intternational Journal of Adaptive Control and Signal Processing* **9**, 199–212.

Messner, W., R. Horowitz, W.W. Kao and M. Boa (1991). A new adaptive learning rule. *IEEE Trans. of Automatic Control* **36**(2), 188–197.

Moore, K. L. (1993). *Iterative learning control for deterministic systems*. Vol. 22 of *Advances in Industrial Control*. Springer-Verlag. London.

Moore, K. L. (1998). Iterative learning control – an expository overview. *Applied & Computational Controls, Signal Processing, and Circuits* pp. 1–42.

Moore, K. L., M. Dahleh and S. P. Bhattacharyya (1992). Iterative learning control: a survey and new results. *J. of Robotic Systems* **9**(5), 563–594.

Mudgett, D. R. and A. S. Morse (1985). Adaptive stabilizing of systems with unknown high frequence gain. *International Journal of Control* **30**, 549–554.

Nakano, M., T. Inoue, Y. Yamamoto and S. Hara (1989). *Repetitive control*. Vol. 22 of *Environmental and Intelligent Manufacturing Systems Series*. SICE. Tokyo, Japan.

Naniwa, T. and S. Arimoto (1991). Learning control for robot tasks under geometric endpoint constraints. *IEEE Trans. of Robotics and Automation* **11**(3), 432–440.

Narendra, K. S. and K. Parthasarathy (1990). Identification and control of dynamic systems using neural networks. *IEEE Trans. Neural Networks* **1**(1), 4–27.

Nussbaum, R.D. (1983). Some remarks on the conjecture in parameter adaptive control. *Syst. Contr. Lett.* **3**, 243–246.

Oh, S.R., Z. Bien and I.H. Suh (1988). An iterative learning control method with application for the robot manipulator. *IEEE J. of Robotics and Automation* **4**(5), 508–514.

Owens, D.H. and G. Munde (1996). Adaptive iterative learning control. *IEE Colloquium on Adaptive Control (Digest No.1996/139)* **45**(1), 6.

Owens, D.H., E. Rogers and K. Galkowski (1999). Control theory and applications for repetitive processes. *Advances in Control. Highlights of ECC'99* **2**(3), 327–333.

Park, K.-H., Z. Bien and D.-H. Hwang (1998). Design of an iterative learning controller for a class of linear dynamic systems with time delay. In: *IEE Proceedings-Control Theory and Applications*. Vol. 145. pp. 507–512.

Park, Kwang-Hyun and Z. Bien (2000). A generalized iterative learning controller against initial state error. *International Journal of Control* **73**(10), 871–881.

Pecora, L. and T. Carroll (1990). Synchronization in chaotic systems. *Phys. Rev. Lett.* **64**, 821–824.

Pepe, P. and E. I. Verriest (2003). On the stability of coupled delay differential and continuous time difference equations. *IEEE Trans. Contr* **48**(8), 1422–1427.

Poggio, T. and F. Girosi (1990). Networks for approximation and learning. In: *Proc. IEEE*. Vol. 78. pp. 1481–1497.

Pogromsky, A. Y. (1998). Passivity based design of synchronizing systems. *International Journal of Bifurcation and Chaos* **8**, 295–320.

Polycarpou, M. M. (1996). Stable adaptive neural control scheme for nonlinear systems. *IEEE Trans. Automat. Contr.* **41**(3), 447–451.

Porter, B. and S. S. Mohamed (1991*a*). Iterative learning control of partially irregular multivariable plants with initial impulsive action. *International Journal of Systems Science* **22**(3), 447–454.

Porter, B. and S. S. Mohamed (1991*b*). Iterative learning control of partially irregular multivariable plants with initial state shifting. *International Journal of Systems Science* **22**(2), 229–235.

Qu, Z. H. (2002). Robust control of nonlinear systems by estimating time variant uncertainties. *IEEE Trans. Automat. Contr.* **47**(1), 115–121.

Qu, Z. H. and J.-X. Xu (2002). Model based learning control and their comparisons using lyapunov direct method. *Asian Journal of Control* **4**, 99–110. (Special issue on ILC).

Rogers, E. and D. H. Owens (1992). *Stability Analysis for Linear Repetitive Processes*. Vol. 8 of *Second Prentice-Hall International Editions*. Springer-Verlag. Berlin.

Rohrs, C. E., L. S. Valavani, M. Athans and G. Stein (1985). Robustness of continuous-time adaptive control algorithms in the presence of unmodeled dynamics. *IEEE Trans. on Aut. Contr.* **30**, 881–889.

Ryan, E. P. (1991). A universal adaptive stabilizer for a class of nonlinear systems. *Syst. Contr. Lett.* **16**, 209–218.

Saab, S. S. (1994). On the P-type learning control. *IEEE Trans. of Automatic Control* **39**(11), 2298–2302.

Sanner, R. and J.-J. E. Slotine (1992). Gaussian networks for direct adaptive control. *IEEE Trans. Neural Networks* **3**, 837–863.

Seshagiri, S. and H. Khalil (2000). Output feedback control of nonlinear systems using rbf neural networks. *IEEE Trans. Neural Networks* **11**, 69–79.

Sira-Ramirez, H. (1991). Nonlinear p-i controller design for switch mode dc-to-dc power converters. *IEEE Transactions on Circuits System* **38**, 410–417.

Song, Y. X., X. H. Yu, G. R. Chen, J.-X. Xu and Y. P. Tian (2002). Time delayed repetitive learning control for chaotic systems. *International Journal of Bifurcation and Chaos* **12**(5), 1057–1065.

Stanford, D. P. and Jr L. T. Conner (1980). Controllability and stabilizability in multi-pair systems. *SIAM Journal of Control and Optimization* **18**(5), 488–497.

Sugie, T. and T. Ono (1991). An iterative learning control law for dynamical systems. *Automatica* **27**(4), 729–732.

Sun, M. X. and D. W. Wang (2001). Initial condition issues on iterative learning control for non-linear systems with time delay. *International Journal of Systems Science* **32**(11), 1365–1375.

Sun, M. X. and D. W. Wang (2002). Iterative learning control with initial rectifying action. *Automatica* **38**(7), 1177–1182.

Sun, Z. and D. Z. Zheng (2001). On reachability and stabilization of switched linear control systems. *IEEE Transactions on Automatic Control* **46**(2), 291–295.

Suykens, J. A. K., P. F. Curran, J. Vandewalle and L. O. Chua (1997). Robust nonlinear h_∞ synchronization of chaotic lur'e systems. *IEEE Trans. Circ. Syst* **44**, 891–904.

Tan, Y. and J.-X. Xu (2003). Learning based nonlinear internal model control. In: *Proceedings of IEEE 2003 American Control Conference*. pp. 3009–3013.

Tayebi, Abdelhamid (2004). Adaptive iterative learning control for robot manipulators. *Automatica* **40**(7), 1195–1203.

Vecchio, D. Del, R. Marino and P. Tomei (2003). Adaptive learning control for feedback linearizable systems. *European Journal of Control* **9**, 479–492.

Walter, G. G. (1995). Pointwise convergence of wavelet expansions. *J. Approx. Theory* **80**, 108–118.

Wang, C. and S. S. Ge (2001*a*). Adaptive synchronization of uncertain chaotic systems via backstepping design. *Chaos, Solitons and Fractals* **12**, 1199–1206.

Wang, C. and S. S. Ge (2001*b*). Synchronization of two uncertain chaotic systems via adaptive backstepping. *International Journal of Bifurcation and Chaos* **11**, 1743–1751.

Wang, D. (1998). Convergence and robust of discrete time nonlinear systems with iterative learning control. *Automatica* **34**(11), 1445–1448.

Wang, J. and X. H. Wang (1998). Parametric adaptive control in nonlinear dynamical systems. *International Journal of Bifurcation and Chaos* **8**(11), 2215–2223.

Williams, S. M. and R. G. Hoft (1991). Adaptive frequency domain control of ppm switched power line conditioner. *IEEE Trans. Power Electron* **6**, 665–670.

Wu, C. W., T. Yang and L. O. Chua (1996). On adaptive synchronization and control of nonlinear dynamical systems. *International Journal of Bifurcation and Chaos* **6**, 455–471.

Xu, J.-X. (1997*a*). Analysis of iterative learning control for a class of nonlinear discrete-time systems. *Automatica* **33**, 1905–1907.

Xu, J.-X. (1997*b*). Direct learning of control efforts for trajectories with different magnitude scales. *Automatica* **33**, 2191–2195.

Xu, J.-X. (1998). Direct learning of control efforts for trajectories with different time scales. *IEEE Transactions on Automatic Control* **43**, 1027–1030.

Xu, J.-X. (2002*a*). The frontiers of iterative learning control – part I. *Journal of Systems, Control and Information* **46**, 63–73.

Xu, J.-X. (2002*b*). The frontiers of iterative learning control – part II. *Journal of Systems, Control and Information*.

Xu, J. X. (2004). A new pointwise adaptive control approach for time-varying parameters with known periodicity. *IEEE Transactions on Automatic Control* **49**, 579–583.

Xu, J.-X. and J. Xu (2002). Iterative learning control for non-uniform trajectory tracking problems. In: *In Proceedings of 15th World Congress of IFAC*. Barcelona, Spain.

Xu, J. X. and R. Yan (2005). On initial conditions in iterative learning control. *IEEE Transactions on Automatic Control* **50**, 1349–1354.

Xu, J.-X. and V. Badrinath (2000). Adaptive robust iterative learning control with dead zone scheme. *Automatica* **36**, 91–99.

Xu, J.-X. and Y. Tan (2000). A composite energy function based sub–optimal learning control approach for nonlinear systems with time–varying parametric uncertainties. In: *In Proceedings of the 39th IEEE Conference on Design and Control*. Sydney, Australia. pp. 3837–3842. (To appear IEEE Trans. Automatic Control).

Xu, J.-X. and Y. Tan (2001). A suboptimal learning control scheme for nonlinear systems with time-varying parametric uncertainties. *Journal of Optimal Control – Applications and Theory*.

Xu, J.-X. and Y. Tan (2002*a*). A composite energy function-based learning control approach for nonlinear systems with time-varying parametric uncertainties. *IEEE Transactions on Automatic Control* **47**, 1940–1945.

Xu, J.-X. and Y. Tan (2002*b*). On the p-type and newton-type ILC schemes for dynamic systems with non-affine-in-input factors. *Automatica* **38**, 1237–1242.

Xu, J.-X. and Y. Tan (2002*c*). On the p-type and newton-type ilc schemes for dynamic systems with non-affine-in-input factors. *Automatica* **38**, 1237–1242.

Xu, J.-X. and Y. Tan (2003). *Linear and Nonlinear Iteration Learning Control*. Vol. 291 of *Lecture Notes in Control and Information Sciences*. Springer-Verlag. ISBN 3–540–40173–3.

Xu, J.-X., V. Badrinath and Z. H. Qu (2000). Robust learning control for robotic manipulators with an extension to a class of non–linear systems. *International Journal of Control* **73**, 858–870.

Ye, H., A. N. Michel and L. Hou (1998). Stability theory for hybrid dynamical systems. *IEEE Transactions on Automatic Control* **43**, 461–474.

Ye, X. D. and J.P. Jiang (1998). Adaptive nonlinear design without a priori knowledge of control directions. *IEEE Trans. Automat. Contr.* **43**(11), 1617–1621.

Yoshizawa, T. (1966). *Stability theory by Liapunov's second method*. Mathematical Society of Japan. Tokyo, Japan.

Yoshizawa, T. (1975). *Stability theory by Liapunov's second method*. Mathematical Society of Japan. Tokyo, Japan.

Yu, X. H. and J.-X. Xu (2000). *Advances in Variable Structure Systems – Analysis, Integration and Applications*. World Scientific. Singapore. ISBN 981–02–4464–9, (Hardcover).

Yu, Xinhuo and Yanxing Song (2001). Chaos synchronization via controlling partial state of chaotic systems. *International Journal of Bifurcation and Chaos* **11**, 1737–1741.

Zhang, B.S., N.A. Jalel and J.R. Leigh (1994). Application of learning control methods to a fed-batch fermentation process. *IEE Conference Publication* **1**(389), 624–628.

Zhang, Huaizhou, Huashu Qin and Guanrong Chen (1998). Adaptive control of chaotic systems with uncertainties. *International Journal of Bifurcation and Chaos* **8**(10), 2041–2046.

Zhang, T., S. S. Ge and C. C. Hang (2000). Stable adaptive control for a class of nonlinear systems using a modified lyapunov function. *IEEE Trans. on Aut. Contr.* **45**(1), 129–132.

Zheng, Z. H., T. R. Ding, W. Z. Huang and Z. X. Dong (1991). *Qualitative Theory of Differential Equations*. American Mathematical Society. Providence, Rhode Island.

Zilouchian, A. (1994). An iterative learning control technique for a dual arm robotic system. In: *Proceedings of the 1994 IEEE International Conference on Robotics and Automation*. Vol. 4. San Diego, CA. pp. 1528–1533.

Appendix A

A.1 Proof of Lemma 2.1

Note that

$$
\begin{aligned}
\Phi\Gamma &= \begin{bmatrix} \boldsymbol{\phi}_1 \\ \vdots \\ \boldsymbol{\phi}_n \end{bmatrix} \begin{bmatrix} \boldsymbol{\gamma}_1 & \cdots & \boldsymbol{\gamma}_n \end{bmatrix} \\
&= \begin{bmatrix} \boldsymbol{\phi}_1\boldsymbol{\gamma}_1 & \cdots & \boldsymbol{\phi}_1\boldsymbol{\gamma}_n \\ \vdots & \ddots & \vdots \\ \boldsymbol{\phi}_n\boldsymbol{\gamma}_1 & \cdots & \boldsymbol{\phi}_n\boldsymbol{\gamma}_n \end{bmatrix} = \begin{bmatrix} \boldsymbol{\gamma}_1^T\boldsymbol{\phi}_1^T & \cdots & \boldsymbol{\gamma}_n^T\boldsymbol{\phi}_1^T \\ \vdots & \ddots & \vdots \\ \boldsymbol{\gamma}_1^T\boldsymbol{\phi}_n^T & \cdots & \boldsymbol{\gamma}_n^T\boldsymbol{\phi}_n^T \end{bmatrix} \\
&= \begin{bmatrix} \boldsymbol{\gamma}_1^T \\ \mathbf{0} \\ \vdots \\ \mathbf{0} \end{bmatrix} \begin{bmatrix} \boldsymbol{\phi}_1^T & \mathbf{0} & \cdots & \mathbf{0} \end{bmatrix} + \cdots + \begin{bmatrix} \boldsymbol{\gamma}_n^T \\ \mathbf{0} \\ \vdots \\ \mathbf{0} \end{bmatrix} \begin{bmatrix} \mathbf{0} & \mathbf{0} & \cdots & \boldsymbol{\phi}_1^T \end{bmatrix} \\
&\quad \vdots \\
&\quad + \begin{bmatrix} \mathbf{0} \\ \mathbf{0} \\ \vdots \\ \boldsymbol{\gamma}_1^T \end{bmatrix} \begin{bmatrix} \boldsymbol{\phi}_n^T & \mathbf{0} & \cdots & \mathbf{0} \end{bmatrix} + \cdots + \begin{bmatrix} \mathbf{0} \\ \mathbf{0} \\ \vdots \\ \boldsymbol{\gamma}_n^T \end{bmatrix} \begin{bmatrix} \mathbf{0} & \mathbf{0} & \cdots & \boldsymbol{\phi}_n^T \end{bmatrix}.
\end{aligned}
$$

According to the definition of Γ^{jk} and Φ^{jk} in Lemma 3.1, the proof is completed.

A.2 Proof of Lemma 2.2

Using the elementary transformation of exchanging rows, we can transform the matrix R into the following form:

$$\tilde{R} = \begin{bmatrix} R_{11} & R_{12} \\ \vdots & \vdots \\ R_{j1} & R_{j2} \\ \vdots & \vdots \\ R_{n1} & R_{n2} \end{bmatrix}, \tag{A.1}$$

where

$$\begin{aligned}
R_{11} &= \begin{bmatrix} \mathbf{d}_{1,1}^T & \cdots & \mathbf{d}_{n,1}^T & \cdots & \mathbf{0} & \cdots & \mathbf{0} \\ \vdots & \ddots & \vdots & \vdots & \vdots & \ddots & \vdots \\ \mathbf{d}_{1,N}^T & \cdots & \mathbf{d}_{n,N}^T & \cdots & \mathbf{0} & \cdots & \mathbf{0} \end{bmatrix}, \\
R_{12} &= \begin{bmatrix} \mathbf{e}_{1,1}^T & \cdots & \mathbf{e}_{n,1}^T & \cdots & \mathbf{0} & \cdots & \mathbf{0} \\ \vdots & \ddots & \vdots & \vdots & \vdots & \ddots & \vdots \\ \mathbf{e}_{1,N}^T & \cdots & \mathbf{e}_{n,N}^T & \cdots & \mathbf{0} & \cdots & \mathbf{0} \end{bmatrix}, \\
&\vdots \\
R_{n1} &= \begin{bmatrix} \mathbf{0} & \cdots & \mathbf{0} & \cdots & \mathbf{d}_{1,1}^T & \cdots & \mathbf{d}_{n,1}^T \\ \vdots & \ddots & \vdots & \vdots & \vdots & \ddots & \vdots \\ \mathbf{0} & \cdots & \mathbf{0} & \cdots & \mathbf{d}_{1,N}^T & \cdots & \mathbf{d}_{n,N}^T \end{bmatrix}, \\
R_{n2} &= \begin{bmatrix} \mathbf{0} & \cdots & \mathbf{0} & \cdots & \mathbf{e}_{1,1}^T & \cdots & \mathbf{e}_{n,1}^T \\ \vdots & \ddots & \vdots & \vdots & \vdots & \ddots & \vdots \\ \mathbf{0} & \cdots & \mathbf{0} & \cdots & \mathbf{e}_{1,N}^T & \cdots & \mathbf{e}_{n,N}^T \end{bmatrix}.
\end{aligned}$$

It is clear that the singularity of the matrix $\tilde{R} \in Nn \times Nn$ is equivalent to the singularity of the matrix $R_1 \in N \times N$. Since the elementary transformation of matrix does not change the rank for the matrix, the rank of the matrix R is equivalent to the rank of the matrix R_1.

A.3 Proof of Proposition 6.1

Choose Lyapunov functional

$$V_0(t) = \frac{1}{2}e_0^2(t) + \frac{1}{2}\int_0^t \phi_0^2(\tau)d\tau. \tag{A.2}$$

The upper right hand derivative of V_0 is

$$\begin{aligned}\dot{V}_0 &= e_0\dot{e}_0 + \frac{1}{2}\phi_0^2 \\ &= -ke_0^2 - \phi_0\xi_0 e_0 + \frac{1}{2}\phi_0^2.\end{aligned}$$

Noticing that $\hat{\theta}_0 = -\xi_0 e_0$, $\dot{V}_0$ becomes

$$\begin{aligned}\dot{V}_0 &= -ke_0^2 + \phi_0\hat{\theta}_0 + \frac{1}{2}\phi_0^2 \\ &= -ke_0^2 - \frac{1}{2}\phi_0^2 + \phi_0\theta.\end{aligned}$$

Using Young's inequality, for any $c > 0$ we have $\phi_0\theta \leq c\phi_0^2 + \frac{1}{4c}\theta^2$. Let $0 < c < \frac{1}{2}$,

$$\dot{V}_0 \leq -ke_0^2 - (\frac{1}{2} - c)\phi_0^2 + \frac{1}{4c}\theta^2.$$

Since $\theta(t) \in \mathcal{C}[0,\ T]$, there exists a finite bound $\theta_m \geq \theta(t)$ for any $t \in [0,\ T]$. Thus $\dot{V}_0$ is negative definite outside the region

$$\{(e_0, \phi_0) \in \mathcal{D} \mid ke_0^2 + (\frac{1}{2} - c)\phi_0^2 \leq \frac{1}{4c}\theta_m^2\}$$

which specifies the bound of $V_0(t)$ in the finite interval $[0,\ T]$. The boundedness of $V_0(t)$ implies the boundedness of e_0, in the sequel the boundedness of x_0, ξ_0, and $\hat{\theta}_0 = -\xi_0 e_0$. □

A.4 Proof of Theorem 6.1

Note that conditions a)-c) are special cases of the condition d), thus we need only to consider the condition d). We will prove this property by the Mathematical Induction method.

Define the following Lyapunov functional

$$V(e_i, \phi_i, \phi_{i-1}, t) = \frac{1}{2}e_i^2(t) + \frac{1}{2}\int_0^t \phi_i^2(\tau)d\tau + \frac{1}{2}\int_t^T \phi_{i-1}^2(\tau)d\tau. \tag{A.3}$$

The upper right hand derivative of $V(e_i, \phi_i, \phi_{i-1}, t)$ is

$$\dot{V}(e_i, \phi_i, \phi_{i-1}, t) = e_i\dot{e}_i + \frac{1}{2}(\phi_i^2 - \phi_{i-1}^2). \tag{A.4}$$

Substituting the closed-loop error dynamics (6.6), the first term on the right hand side of (A.4) is

$$e_i\dot{e}_i = -\phi_i\xi_i e_i - ke_i^2. \tag{A.5}$$

Next substituting the parametric learning law (6.5) into the second term on the right hand side of (A.4), using the relations $(a-b)^2 - (a-c)^2 = -2(a-b)(b-c) - (b-c)^2$ and the property $(\theta - \hat{\theta})^2 \geq (\theta - \mathrm{proj}(\hat{\theta}))^2$ for any $\hat{\theta}$, we have

$$\begin{aligned}
\frac{1}{2}(\phi_i^2 - \phi_{i-1}^2) &= \frac{1}{2}[(\theta - \hat{\theta}_i)^2 - (\theta - \hat{\theta}_{i-1})^2] \\
&\leq \frac{1}{2}[(\theta - \hat{\theta}_i)^2 - (\theta - \mathrm{proj}(\hat{\theta}_{i-1}))^2] \\
&= -(\theta - \hat{\theta}_i)(\hat{\theta}_i - \mathrm{proj}(\hat{\theta}_{i-1})) - \frac{1}{2}(\hat{\theta}_i - \mathrm{proj}(\hat{\theta}_{i-1}))^2 \\
&= \phi_i\xi_i e_i - \frac{1}{2}\xi_i^2 e_i^2.
\end{aligned} \tag{A.6}$$

Clearly $\phi_i\xi e_i$ appears in (A.5) and (A.6) with opposite signs. Therefore, the upper right hand derivative of $V(e_i, \phi_i, \phi_{i-1}, t)$ is

$$\dot{V}(e_i, \phi_i, \phi_{i-1}, t) = -ke_i^2 - \frac{1}{2}\xi_i^2 e_i^2 < 0. \tag{A.7}$$

Integrating the derivative of V, using the negativeness of $\dot{V}$, the boundedness of e_i and $\hat{\theta}_i$ can be derived if $V(e_i(0), \phi_i(0), \phi_{i-1}(0))$ is bounded, i.e.

$$\begin{aligned}
V(e_i(t), \phi_i(t), \phi_{i-1}(t), t) &= V(e_i(0), \phi_i(0), \phi_{i-1}(0), 0) + \int_0^t \dot{V}dt \\
&\leq V(e_i(0), \phi_i(0), \phi_{i-1}(0), 0).
\end{aligned} \tag{A.8}$$

Note that

$$V(e_i(0), \phi_i(0), \phi_{i-1}(0), 0) = \frac{1}{2}e_i^2(0) + \frac{1}{2}\int_0^T \phi_{i-1}^2(\tau)d\tau$$

and $e_i(0)$ is always bounded by the initial condition d).

Let us look at the first iteration $i = 1$,

$$V(e_1(0), \phi_1(0), \phi_0(0), 0) = \frac{1}{2}e_1^2(0) + \frac{1}{2}\int_0^T \phi_0^2(\tau)d\tau$$

is bounded because $\phi_0(t)$ is bounded according to Proposition 1. In the sequel $V(e_1(t), \phi_1(t), \phi_0(t), t) \le V(e_1(0), \phi_1(0), \phi_0(0), 0)$ is bounded. From the parametric learning law (6.5), the boundedness of e_1 warrants the boundedness of $\hat{\theta}_1$.

Now assume that $(e_{i-1}, \hat{\theta}_{i-1})$ are bounded for all $t \in [0,\ T]$, so is $V(e_i(0), \phi_i(0), \phi_{i-1}(0), 0)$. From (A.8), $V(e_i(t), \phi_i(t), \phi_{i-1}(t), t)$ is bounded. Similarly, from the boundedness of e_i and the parametric learning law (6.5) we can derive the boundedness of $\hat{\theta}_i$. By the Mathematical Induction, the quantities $(e_i, \hat{\theta}_i)$ are bounded for any $i \ge 0$.

□

A.5 Proof of Proposition 6.2

The difference between V_i and V_{i-1} is

$$\begin{aligned} \Delta V_i &= V_i - V_{i-1} \\ &= \frac{1}{2}e_i^2 + \int_0^t (\phi_i^2 - \phi_{i-1}^2)d\tau - \frac{1}{2}e_{i-1}^2. \end{aligned} \tag{A.9}$$

Substituting the control law (6.4) and the error dynamics (6.6), the first term on the right hand side of (A.9) is

$$\begin{aligned} \frac{1}{2}e_i^2 &= \int_0^t e_i \dot{e}_i d\tau + \frac{1}{2}e_i^2(0) \\ &= \int_0^t (-\phi_i \xi_i e_i - k e_i^2)d\tau + \frac{1}{2}e_i^2(0). \end{aligned}$$

Similarly as (A.6), the second term on the right hand side of (A.9) can be expressed as

$$\frac{1}{2}\int_0^t (\phi_i^2 - \phi_{i-1}^2)d\tau \le \int_0^t (\phi_i \xi_i e_i - \frac{1}{2}\xi_i^2 e_i^2)d\tau.$$

Therefore, the difference becomes

$$\Delta V_i \leq -\int_0^t k e_i^2 d\tau - \frac{1}{2}\int_0^t \xi_i^2 e_i^2 d\tau - \frac{1}{2}e_{i-1}^2(t) + \frac{1}{2}e_i^2(0). \tag{A.10}$$

Applying (A.10) repeatedly we have

$$\begin{aligned} V_i(t) &= V_0(t) + \sum_{j=1}^{i} \Delta V_j \\ &\leq V_0(t) + \frac{1}{2}\sum_{j=1}^{i} e_j^2(0) - \sum_{j=1}^{i}\int_0^t k e_j^2 d\tau - \frac{1}{2}\sum_{j=1}^{i-1} e_j^2(t), \end{aligned}$$

consequently

$$\lim_{i\to\infty} V_i(t) \leq V_0(t) + \lim_{i\to\infty}\frac{1}{2}\sum_{j=1}^{i} e_j^2(0) - \lim_{i\to\infty}\sum_{j=1}^{i}\int_0^t k e_j^2 d\tau - \lim_{i\to\infty}\frac{1}{2}\sum_{j=1}^{i-1} e_j^2(t).$$

□

A.6 Adaptive Robust Control Design

Consider the following 2nd order cascaded dynamic system

$$\begin{aligned} \dot{x}_1 &= x_2 + \eta_1(t, \mathbf{x}_1), \\ \dot{x}_2 &= u + \eta_2(t, \mathbf{x}), \end{aligned} \tag{A.11}$$

Define new coordinates $z_1 = x_1 - x_{r,1}$ and $z_2 = x_2 - u_1$, where the fictitious control is

$$u_1 = -(\alpha_1 + q_1)z_1 + x_{r,2} - \mathcal{S}(\hat{\beta}_1 z_1)\hat{\beta}_1 \tag{A.12}$$

with $q_1 > 0$. $\hat{\beta}_1$ is the estimation of β_1 and β_1 is the upper bound of η_{r_1}.

Design

$$\dot{\hat{\beta}}_1 = |z_1| + \left|\frac{\partial u_1}{\partial x_1} z_2\right| - \gamma_1 \hat{\beta}_1, \tag{A.13}$$

where $\gamma_1 > 0$ is a damping coefficient.

Design the actual controller

$$u = f_2 - z_1 - q_2 z_2 - \mathcal{S}(\bar{\alpha}_2 z_2)\bar{\alpha}_2 - \mathcal{S}\left(\frac{\partial u_1}{\partial x_1}\hat{\beta}_1 z_2\right)\frac{\partial u_1}{\partial x_1}\hat{\beta}_1 - \mathcal{S}(\hat{\beta}_2 z_2)\hat{\beta}_2 \tag{A.14}$$

with $q_2 > 0$,

$$f_2 = \frac{\partial u_1}{\partial t} + \frac{\partial u_1}{\partial x_{r,1}} x_{r,2} + \frac{\partial u_1}{\partial x_{r,2}} s(t, \mathbf{x}_r, r) + \frac{\partial u_1}{\partial \hat{\beta}_1}\dot{\hat{\beta}}_1 + \frac{\partial u_1}{\partial x_1} x_2,$$

and

$$\bar{\alpha}_2 = \left(\alpha_2 + \alpha_1 \left|\frac{\partial u_1}{\partial x_1}\right|\right)|\Delta x_1| + \alpha_2^{'}|\Delta x_2|.$$

The updating law is

$$\dot{\hat{\beta}}_2 = |z_2| - \gamma_2 \hat{\beta}_2, \tag{A.15}$$

with $\gamma_2 > 0$.

Theorem A.1. *For system (A.11), the control law (A.14), the adaptation law (A.13) and (A.15) guarantee the finiteness of z_1 and z_2 in the large, and the tracking error bound of z_1 is*

$$|z_1| \le \sqrt{\frac{6\delta + \gamma_1 \beta_1^2 + \gamma_2 \beta_2^2}{2q_1}}. \tag{A.16}$$

Proof. The proof consists of two steps.

Step 1.

From (A.11), we have

$$\begin{aligned} \dot{z}_1 &= \dot{x}_1 - \dot{x}_{r,1} \\ &= x_2 + \eta_1 - x_{r,2} \\ &= z_2 + u_1 + \eta_1 - x_{r,2}. \end{aligned} \tag{A.17}$$

Substituting the fictitious control u_1 (A.12) into (A.17) yields

$$\begin{aligned} \dot{z}_1 &= z_2 - (\alpha_1 + q_1)z_1 + \eta_1 - \mathcal{S}(\hat{\beta}_1 z_1)\hat{\beta}_1 \\ &= z_2 - (\alpha_1 + q_1)z_1 - \mathcal{S}(\hat{\beta}_1 z_1)\hat{\beta}_1 + (\eta_1 - \eta_{r,1}) + \eta_{r,1}. \end{aligned} \tag{A.18}$$

Define a Lyapunov function candidate below

$$V_1 = \frac{1}{2}z_1^2 + \frac{1}{2}(\beta_1 - \hat{\beta}_1)^2. \tag{A.19}$$

Using (A.18), adaptation law (A.13) and Property 8.1, the derivative of V_1 is

$$\begin{aligned} \dot{V}_1 &= z_1\dot{z}_1 - (\beta_1 - \hat{\beta}_1)\dot{\hat{\beta}}_1 \\ &= z_1[z_2 - (\alpha_1 + q_1)z_1 - \mathcal{S}(\hat{\beta}_1 z_1)\hat{\beta}_1 + (\eta_1 - \eta_{r,1}) + \eta_{r,1}] - (\beta_1 - \hat{\beta}_1)\dot{\hat{\beta}}_1 \\ &= z_1 z_2 - (\alpha_1 + q_1)z_1^2 + (\eta_1 - \eta_{r,1})z_1 - \mathcal{S}(\hat{\beta}_1 z_1)\hat{\beta}_1 z_1 + \eta_{r,1}z_1 - (\beta_1 - \hat{\beta}_1)\dot{\hat{\beta}}_1 \\ &\leq z_1 z_2 - q_1 z_1^2 - \mathcal{S}(\hat{\beta}_1 z_1)\hat{\beta}_1 z_1 + \beta_1|z_1| - (\beta_1 - \hat{\beta}_1)\dot{\hat{\beta}}_1 \\ &= z_1 z_2 - q_1 z_1^2 - \mathcal{S}(\hat{\beta}_1 z_1)\hat{\beta}_1 z_1 + \hat{\beta}_1|z_1| - \hat{\beta}_1|z_1| + \beta_1|z_1| - (\beta_1 - \hat{\beta}_1)\dot{\hat{\beta}}_1 \\ &\leq z_1 z_2 - q_1 z_1^2 + |\hat{\beta}_1 z_1|[1 - |\mathcal{S}(\hat{\beta}_1 z_1)|] - (\beta - \hat{\beta}_1)(\dot{\hat{\beta}}_1 - |z_1|) \\ &\leq z_1 z_2 - q_1 z_1^2 + \delta - (\beta_1 - \hat{\beta}_1)(\dot{\hat{\beta}}_1 - |z_1|). \end{aligned} \tag{A.20}$$

Step 2.

From (A.11) and (A.12), we have

$$\begin{aligned} \dot{z}_2 &= \dot{x}_2 - \dot{u}_1 \\ &= u + \eta_2 - \left(\frac{\partial u_1}{\partial t} + \frac{\partial u_1}{\partial x_1}\dot{x}_1 + \frac{\partial u_1}{\partial x_{r,1}}x_{r,2} + \frac{\partial u_1}{\partial x_{r,2}}s(t, \mathbf{x}_r, r) + \frac{\partial u_1}{\partial \hat{\beta}_1}\dot{\hat{\beta}}_1\right) \\ &= u - \left(\frac{\partial u_1}{\partial t} + \frac{\partial u_1}{\partial x_{r,1}}x_{r,2} + \frac{\partial u_1}{\partial x_{r,2}}s(t, \mathbf{x}_r, r) + \frac{\partial u_1}{\partial \hat{\beta}_1}\dot{\hat{\beta}}_1\right) + \eta_2 - \frac{\partial u_1}{\partial x_1}(x_2 + \eta_1) \\ &= u - f_2 - g_1\eta_{r,1} + \eta_{r,2} \\ &\quad -\frac{\partial u_1}{\partial x_1}(\eta_1 - \eta_{r,1}) + [\eta_2 - \eta_2(t, x_{r,1}, x_2)] + [\eta_2(t, x_{r,1}, x_2) - \eta_{r,2}] \\ &= u - f_2 - g_1\eta_{r_1} + \eta_{r_2} - \frac{\partial u_1}{\partial x_1}(\eta_1 - \eta_{r,1}) \\ &\quad +[\eta_2 - \eta_2(t, x_{r,1}, x_2)] + [\eta_2(t, x_{r,1}, x_2) - \eta_{r,2}], \end{aligned} \tag{A.21}$$

where $g_1 = \frac{\partial u_1}{\partial x_1}$ and

$$f_2 = \frac{\partial u_1}{\partial t} + \frac{\partial u_1}{\partial x_{r,1}} x_{r,2} + \frac{\partial u_1}{\partial x_{r,2}} s(t, \mathbf{x}_r, r) + \frac{\partial u_1}{\partial \hat{\beta}_1} \dot{\hat{\beta}}_1 + \frac{\partial u_1}{\partial x_1} x_2$$

are known.

Substituting (A.14) into (A.21) yields

$$\begin{aligned} \dot{z}_2 &= -z_1 - q_2 z_2 - \mathcal{S}\left(\hat{\beta}_1 g_1 z_2\right) \hat{\beta}_1 g_1 - g_1 \eta_{r_1} - \mathcal{S}(\hat{\beta}_2 z_2)\hat{\beta}_2 + \eta_{r_2} - \mathcal{S}(\bar{\alpha}_2 z_2)\bar{\alpha}_2 \\ &\quad - \frac{\partial u_1}{\partial x_1}(\eta_1 - \eta_{r,1}) + [\eta_2 - \eta_2(t, x_{r,1}, x_2)] + [\eta_2(t, x_{r,1}, x_2) - \eta_{r,2}] \end{aligned} \tag{A.22}$$

Define the Lyapunov functional below

$$V_2 = V_1 + \frac{1}{2} z_2^2 + \frac{1}{2}(\beta_2 - \hat{\beta}_2)^2. \tag{A.23}$$

The upper right hand derivative of V_2 is

$$\dot{V}_2 = \dot{V}_1 + z_2 \dot{z}_2 - (\beta_2 - \hat{\beta}_2)\dot{\hat{\beta}}_2. \tag{A.24}$$

Using (A.22) and Property 8.1, we have

$$\begin{aligned} z_2 \dot{z}_2 &= -z_1 z_2 - q_2 z_2^2 - \mathcal{S}(\hat{\beta}_1 g_1 z_2)\hat{\beta}_1 g_1 z_2 - g_1 \eta_{r_1} z_2 - \mathcal{S}(\hat{\beta}_2 z_2)\hat{\beta}_2 z_2 \\ &\quad + \eta_{r_2} z_2 - \mathcal{S}(\bar{\alpha}_2 z_2)\bar{\alpha}_2 z_2 - \frac{\partial u_1}{\partial x_1}(\eta_1 - \eta_{r,1}) z_2 \\ &\quad + [\eta_2 - \eta_2(t, x_{r,1}, x_2)] z_2 + [\eta_2(t, x_{r,1}, x_2) - \eta_{r,2}] z_2 \\ &\leq -z_1 z_2 - q_2 z_2^2 - \mathcal{S}(\hat{\beta}_1 g_1 z_2)\hat{\beta}_1 g_1 z_2 + \beta_1 |g_1 z_2| - \mathcal{S}(\hat{\beta}_2 z_2)\hat{\beta}_2 z_2 + \beta_2 |z_2| \\ &\quad + \bar{\alpha}_2 |z_2| - \mathcal{S}(\bar{\alpha}_2 z_2)\bar{\alpha}_2 z_2 \\ &\leq -z_1 z_2 - q_2 z_2^2 - \mathcal{S}(\hat{\beta}_1 g_1 z_2)\hat{\beta}_1 g_1 z_2 + \hat{\beta}_1 |g_1 z_2| - \hat{\beta}_1 |g_1 z_2| + \beta_1 |g_1 z_2| \\ &\quad - \mathcal{S}(\hat{\beta}_2 z_2)\hat{\beta}_2 z_2 + \hat{\beta}_2 |z_2| - \hat{\beta}_2 |z_2| + \beta_2 |z_2| + \delta \\ &\leq -z_1 z_2 - q_2 z_2^2 + (\beta_1 - \hat{\beta}_1)|g_1 z_2| + (\beta_2 - \hat{\beta}_2)|z_2| + 3\delta \end{aligned} \tag{A.25}$$

Substituting (A.20) and (A.25) into (A.24) yields

$$\begin{aligned} \dot{V}_2 &\leq -q_1 z_1^2 - q_2 z_2^2 + 3\delta - (\beta_1 - \hat{\beta}_1)(\dot{\hat{\beta}}_1 - |z_1| - |g_1 z_2|) \\ &\quad - (\beta_2 - \hat{\beta}_2)(\dot{\hat{\beta}}_2 - |z_2|) \end{aligned} \tag{A.26}$$

Note the adaptation law (A.13) and learning law (A.15), we have

$$\begin{aligned}\dot{V}_2 &\leq -q_1 z_1^2 - q_2 z_2^2 + 3\delta + \gamma_1 \hat{\beta}_1(\beta_1 - \hat{\beta}_1) + \gamma_2 \hat{\beta}_2(\beta_2 - \hat{\beta}_2) \\ &\leq -q_1 z_1^2 - q_2 z_2^2 + 3\delta - \gamma_1(\frac{1}{2}\hat{\beta}_1^2 - \hat{\beta}_1\beta_1) - \gamma_2(\frac{1}{2}\hat{\beta}_2^2 - \hat{\beta}_2\beta_2) \\ &= -q_1 z_1^2 - q_2 z_2^2 - \frac{\gamma_1}{2}(\hat{\beta}_1 - \beta_1)^2 - \frac{\gamma_2}{2}(\hat{\beta}_2 - \beta_2)^2 \\ &\quad +3\delta + \frac{\gamma_1}{2}\beta_1^2 + \frac{\gamma_2}{2}\beta_2^2 \end{aligned} \tag{A.27}$$

The following proof is the same as that of Theorem 8.4. □

A.7 Proof of Property 9.1

The upper right-hand derivative of the integral is

$$\lim_{\Delta t \to 0^+} sup \frac{\int_{t+\Delta t-T}^{t+\Delta t} \theta^2(\tau)d\tau - \int_{t-T}^{t} \theta^2(\tau)d\tau}{\Delta t}. \tag{A.28}$$

Note the fact

$$\int_{t+\Delta t-T}^{t+\Delta t} \theta^2(\tau)d\tau = \int_{t-T+\Delta t}^{t-T} \theta^2(\tau)d\tau + \int_{t-T}^{t} \theta^2(\tau)d\tau + \int_{t}^{t+\Delta t} \theta^2(\tau)d\tau$$

and substitute into (A.28), we have

$$\begin{aligned}&\lim_{\Delta t \to 0^+} sup \frac{\int_{t+\Delta t-T}^{t+\Delta t} \theta^2(\tau)d\tau - \int_{t-T}^{t} \theta^2(\tau)d\tau}{\Delta t} \\ &= \lim_{\Delta t \to 0^+} sup \frac{\int_{t}^{t+\Delta t} \theta^2(\tau)d\tau - \int_{t-T}^{t-T+\Delta t} \theta^2(\tau)d\tau}{\Delta t} \\ &= \theta^2(t) - \theta^2(t-T). \end{aligned} \tag{A.29}$$

A.8 Proof of Property 9.2

From Property 9.1, the upper right-hand derivative of

$$\int_{t-T}^{t} \tilde{\theta}^2(\tau)d\tau$$

is

$$\tilde{\theta}^2(t) - \tilde{\theta}^2(t-T).$$

Using the relation (9.4),

$$\begin{aligned}
\tilde{\theta}^2(t-T) &= [\theta(t-T) - \hat{\theta}(t-T)][\theta(t-T) - \hat{\theta}(t-T)] \\
&= [\theta(t) - \hat{\theta}(t) + f(t)][\theta(t) - \hat{\theta}(t) + f(t)] \\
&= \tilde{\theta}^2(t) + 2f(t)\tilde{\theta}(t) + f^2(t). \qquad \text{(A.30)}
\end{aligned}$$

Substituting the above relation into (A.30) yields

$$-2\tilde{\theta}(t)f(t) - f^2(t).$$

Appendix B

Author's Publications

The author has contributed to the following publications:

Journal Publications

1. J.-X. Xu and R. Yan, "Fixed point theorem based iterative learning control for LTV systems with input singularity", *IEEE Transactions on Automatic Control*, Vol. 48, no. 3, pp. 487-492, 2003.

2. J.-X. Xu, R. Yan and Z. H. Guan, "Direct learning control design for a class of linear time-varying switched systems", *IEEE Transactions on Circuits and Systems*, Part I, Vol. 50, no. 8, pp. 1116-1120, 2003.

3. J.-X. Xu and R. Yan, "Iterative learning control design without a priori knowledge of the control direction", *Automatica*, Vol. 40, pp. 1803-1809, 2004.

4. J.-X. Xu, R. Yan and W.N. Zhang, "An algorithm of Melnikov function and application to a chaotic rotor", *SIAM Journal of Scientific Computing*, Vol. 26, no. 5 pp. 1525-1546, 2005.

5. J.-X. Xu and R. Yan, "On Initial Conditions in Iterative Learning Control", *IEEE Transactions on Automatic Control*, Vol 50, no 9, 2005.

6. J.-X. Xu and R. Yan, "Synchronization of Chaotic Systems Via Learning Control", *International Journal of Bifurcation and Chaos*, accepted.

7. J.-X. Xu, W.N. Zhang, Y. J. Pan and R. Yan, "Periodicity of an Implicit Difference Equation with Discontinuity and Its Simulations", *International Journal of Bifurcation and Chaos*, accepted.

8. J.-X. Xu and R. Yan, "Repetitive Learning Control: Existence of Solution, Convergence and Robustification", *IEEE Transactions on Automatic Control*, revised.

9. J.-X. Xu and R. Yan, "Constructive Iterative Learning Control Based on Function Approximation and Wavelet", *IEEE Transactions on Neural Network* , submitted.

10. J.-X. Xu and R. Yan, "Repetitive Learning Control: A Time-delay Approach for Systems with Periodic Components", *SIAM Journal on Control and Optimization*, submitted.

Conference Publications

1. J.-X. Xu, Y. Tan and R. Yan, "On the Existence and Uniqueness of Inverse Mapping for a Class of Dynamical Systems with Volterra Operator". *In Proceedings of the 3rd IEEE International Conference on Control Theory and Application, Singapore South African Council for Automation and Computation*, December, 2001, Pretoria, South Africa.

2. J.-X. Xu and R. Yan, "Fixed point theorem based iterative learning control for LTV systems with input singularity", *In Proceedings of IEEE 2003 American Control Conference*, pp.3655-3660, 2003.

3. J.-X. Xu and R. Yan, "Iterative learning control design without a priori knowledge of the control direction", *In Proceedings of IEEE 2003 American Control Conference*, pp.3661-3666, 2003.

4. J.-X. Xu, R. Yan and Z. H. Guan, "Direct learning control design for a class of linear time-varying switched systems", *In Proceedings of The 4th International Conference on Control Theory and Applications*, pp.466-470, Montreal, Canada, 2003.

5. J.-X. Xu and R. Yan, "An Adaptive Learning Control Approach Based on Constructive Function Approximation", *International Joint Conference Neural Networks*, 2004.

6. J.-X. Xu and R. Yan, "Constructive Iterative Learning Control Based on Function Approximation and Wavelet", *43rd IEEE Conference on Decision and Control*, 2004.

7. J.-X. Xu and R. Yan, "Synchronization of Chaotic Systems Via Learning Control", *ICARCV, 2004* .

8. J.-X. Xu and R. Yan, "On Initial Conditions in Iterative Learning Control", *44rd IEEE Conference on Decision and Control*, accepted, 2005.