ON ITERATIVE LEARNING IN MULTI-AGENT SYSTEMS COORDINATION AND CONTROL

YANG SHIPING

(B.Eng. (Hons.), NUS)

A THESIS SUBMITTED

FOR THE DEGREE OF DOCTOR OF PHILOSOPHY

NUS GRADUATE SCHOOL FOR INTEGRATIVE SCIENCES

AND ENGINEERING

NATIONAL UNIVERSITY OF SINGAPORE

2014

Declaration

I hereby declare that the thesis is my original work and it has been written by me in its entirely. I have duly acknowledged all the sources of information which have been used in the thesis.

This thesis has also not been submitted for any degree in any university previously.

SHIPING YANG

31 July, 2014

Acknowledgments

I would like to express my sincere appreciation to my supervisor Professor Xu Jian-Xin. With his rich experience in research and vast knowledge in learning control, Professor Xu inspired and guided me to the right research direction throughout the four-year PhD program. In our countless discussions, Professor Xu treated me more like a researcher instead of a student by taking my opinions seriously and offering me great research autonomy, which cultivated the independent problem solving ability. Besides, Professor Xu's objective and rigorous attitude towards academic research also influenced my working style. I own a debt of gratitude to him for his excellent supervision.

I would like to take this opportunity to thank my Thesis Advisory Committee, Professor Chen Ben Mei and Professor Chu Delin. Thanks for giving me constructive comments on the research work and also for sharing their life experience with me.

I would also like to thank Dr. Tan Ying for introducing us the concept of iISS, which eventually leads to the key proof idea in Chapter 5.

Special thanks go to NUS Graduate School for Integrative Sciences and Engineering, Electrical and Computer Engineering, and Ministry of Education Singapore. Thanks so much for your support over the years.

I am grateful to my friends in the Control and Simulation Lab. Thanks for your encouragement, friendship, and support. We are not alone on the journey towards PhD.

Lastly, I would like to thank my wife, Ms. Zhang Jiexin, for her love and constant support. Having Jiexin in my life is one of the driving forces to complete the program. Thanks for sharing the best and the worst parts in the past four years.

This thesis is dedicated to my grandma Cai Guoxiu.

Contents

Summary

Multi-agent systems coordination and control problem has been extensively studied by the control community as it has wide applications in practice. For example, the formation control problem, search and rescue by multiple aerial vehicles, synchronization, sensor fusion, distributed optimization, economic dispatch problem in power systems, etc. Meanwhile, many industry processes require both repetitive executions and coordination among several independent entities. This observation motivates the research of multi-agent coordination from iterative learning control (ILC) perspective.

To study multi-agent coordination by ILC, an extra dimension, the iteration domain, is introduced to the problem. In addition, the inherent nature of multi-agent systems such as heterogeneity, information sharing, sparse and intermittent communication, imperfect initial conditions increases the complexity of the problem. Due to these factors, the controller design becomes a challenging problem. This thesis aims at designing learning controllers under various coordination conditions and analyzing the convergence properties. It follows the two main frameworks of ILC, namely contraction-mapping (CM) and composite energy function (CEF) approaches. In the first part, assuming a fixed communication topology and perfect initial conditions, CM based iterative learning controller is developed for multi-agent consensus tracking problem. By using the concept of a graph dependent matrix norm, the convergence conditions are given at the agent level, which depend on a set of eigenvalues that are associated with the communication topology. Next, optimal controller gain design methods are proposed in the sense that the λ-norm of the tracking error converges at the fastest rate, which imposes a tightest bounding function for the actual tracking error in the λ-norm analysis. As the communication is one of the indispensable components of multi-agent coordination, robustness against communication variation is desirable. By utilizing the properties of substochastic matrix, it is shown that under very weak interactions among agents such as uniformly strongly connected graph in the iteration domain, controller convergence can be preserved. Furthermore, in the multi-agent systems each agent is an independent entity. Hence it is difficult to guarantee the perfect initial conditions for all agents in the system. Therefore, it is crucial for the learning algorithm to work under imperfect initial conditions. In this thesis, a PD-type learning rule is developed for the multi-agent setup. The new learning rule facilitates two degree of freedom in the controller design. On the one hand, it ensures the convergence of the controller; on the other hand, it can improve the final tracking control performance. In the second part, the applicability of P-type learning rule to local Lipschitz continuous systems is explored since it is believed that CM based ILC is only applicable to global Lipschitz continuous systems, which restricts its application to limited systems. By combining Lyapunov method and the advantages of CM analysis method, several sufficient conditions in the form of Lyapunov function criteria are developed for ILC convergence, which greatly complements the existing literature. To deal with the general local Lipschitz systems which can be linearly parameterized, CEF based learning rules are developed for multi-agent synchronization problem. The results are first derived for SISO systems, and then generalized to high-order systems. Imperfect initial conditions are considered as well. Finally, a set of distributed learning rules are developed to synchronize networked Lagrangian systems under directed acyclic graph. The inherent properties of Lagrangian systems such as positive definiteness, skew symmetric, and linear in parameter properties, are fully utilized in the controller design to enhance the performance.

List of Figures

Chapter 1

Introduction

1.1 Introduction to Iterative Learning Control

Iterative learning control (ILC) is a memory based intelligent control strategy, which is developed to deal with repeatable control tasks defined on fixed and finite-time intervals. The underlying philosophy mimics the human learning process that practice makes perfect. By synthesizing the control input from the previous control input and tracking error, the controller is able to learn from the past experience and improve the current tracking performance. ILC was initially developed by Arimoto et al. (1984), and has been widely explored by the control community since then (Moore, 1993; Longman, 2000; Norrlof and Gunnarsson, 2002; Xu and Tan, 2003; Bristow et al., 2006; Moore et al., 2006; Wang et al., 2009; Ahn et al., 2007).

Generally speaking there are two main frameworks for ILC, namely contraction-mapping (CM) and composite energy function (CEF) based approaches. CM based iterative learning controller has a very simple structure and it is extremely easy to implement. A correction term in the controller is constructed by the output tracking error. To ensure convergence, an appropriate learning gain can be selected based on the system gradient information instead of accurate dynamic model. As it is a partial model-free control method, CM based ILC is applicable to non-affine in input systems. These features are highly desirable in practice as there are plenty of data available in the industry processes but are lack of accurate system models. CM based ILC has been adopted in many applications, for example X-Y table, chemical batch reactors, laser cutting system, motor control, water heating system, freeway traffic control, wafer manufacturing, and etc (Ahn et al., 2007). Whereas, CM based ILC is only applicable to global Lipschitz continuous (GLC) systems. On the one hand, it is because CM based ILC is an open loop system in the time domain and a closed loop system in the iteration domain. GLC is required by the learning controller in order to rule out the finite escape time phenomenon. On the other hand, GLC is a key assumption to construct a contraction-mapping such that the controller convergence can be proven. In comparison, CEF based ILC, a complementary part of CM based ILC, applies Lyapunov method to design learning rules. It is an effective method to handle local Lipschitz continuous (LLC) systems. However, the system dynamics must be in linear in parameter form and full state information must be available for feedback or nonlinear compensation. As the current state tracking error is used in the feedback, the transient performance is usually better than CM based ILC. CEF based ILC has been applied in satellite trajectory keeping (Ahn et al., 2010) and robotics manipulators control (Tayebi, 2004; Tayebi and Islam, 2006; Sun et al., 2006).

This thesis follows the two main frameworks and investigates the multi-agent coordination problem by ILC.

1.2 Introduction to Multi-agent Systems Coordination

In the past several decades, multi-agent systems coordination and control problems have attracted considerable attention from many researchers of various backgrounds due to their potential applications and cross-disciplinary nature. In particular consensus is an important class of multi-agent systems coordination and control problems (Cao et al., 2013). According to Olfati-Saber et al. (2007), in networks of agents (or dynamic systems), consensus means to reach an agreement regarding certain quantities of interest that are associated with the agents. Depending on the specific applications these quantities could be velocity, position, temperature, orientation, and etc. In a consensus realization, the control action of an agent is generated based on the information received or measured from its neighborhood. Since the control law is a kind of distributed algorithm, it is more robust and scalable compared to centralized control algorithms.

Consensus algorithm is a very simple local coordination rule which can result in very complex and useful behaviors at the group level. For instance, it is widely observed that by adopting such a strategy, a school of fish can improve the chance of survival under the sea (Moyle and Cech, 2003). Many interesting coordination problems have been formulated and solved under the framework of consensus, e.g., distributed sensor fusion (Olfati-Saber et al., 2007), satellite alignment problem (Ren and Beard, 2008), multi-agent formation (Ren et al., 2007), synchronization of coupled oscillators (Ren, 2008a), and optimal dispatch in power systems (Yang et al., 2013). Consensus problem is usually studied in the infinite time horizon, that is the consensus is reached when time tends to infinity. Meanwhile some finite-time convergence algorithms are available (Cortex, 2006; Wang and Hong, 2008; Khoo et al., 2009; Wang and Xiao, 2010; Li et al., 2011). In the existing literature, most consensus algorithms are model-based algorithms. The agent models range from simple single integrator model to complex nonlinear models. Consensus results on single integrators are reported by Jadbabaie et al. (2003); Olfati-Saber and Murray (2004); Moreau (2005); Ren et al. (2007); Olfati-Saber et al. (2007). Double integrators are investigated in Xie and Wang (2005); Hong et al. (2006); Ren (2008b); Zhang and Tian (2009). Results on linear agent models can be found in Xiang et al. (2009); Ma and Zhang (2010); Li et al. (2010); Huang (2011); Wieland et al. (2011). Since the Euler-Lagrangian system can be used to model many practical systems, consensus has been extensively studied by Euler-Lagrangian system. Some representative works are reported by Hou et al. (2009b); Chen and Lewis (2011); Mei et al. (2011); Zhang et al. (2012). Information sharing among agents is one of the indispensable components for consensus seeking. Information sharing can be realized by direct measurement from on board sensors or communication through wireless networks. The information sharing mechanism is usually modeled by graph. For simplicity in the early stage, the communication graph is assumed to be fixed. However, a consensus algorithm, which is insensitive to topology variations, is more desired since many practical conditions can be modeled as time-varying communication, for example, asynchronous updating, communication link failures and creations. As communication among agents is an important topic in multi-agent systems literature, various communication assumptions and consensus results are investigated by researchers (Moreau, 2005; Hatano and Mesbahi, 2005; Tahbaz-Salehi and Jadbabaie, 2008; Zhang and Tian, 2009). An excellent survey paper can be found in Fang and Antsaklis (2006)

1.3 Motivation and Contribution

In practice, there are many tasks requiring both repetitive executions and coordinations among several independent entities. For example, it is useful for a group of satellites to orbit the earth in formation for positioning or monitoring purposes (Ahn et al., 2010). Each satellite orbiting the earth is a periodic task, and the formation task fits perfectly in the ILC framework. Another example is the cooperative transportation of a heavy load by multiple mobile robots (Bai and Wen, 2010; Yufka et al., 2010). In such kind of task implementations, the robots have to maneuver in formation from the very beginning to the destination. Besides, the economic dispatch problem in power systems (Xu and Yang, 2013; Yang et al., 2013) and the formation control for ground vehicles with nonholonomic constraints (Xu et al., 2011) also fall in this category. These observations motivate the study of multi-agent coordination control from the perspective of ILC.

The objective of the thesis is to design and analyze iterative learning controllers for multi-agent systems which perform collaborative tracking tasks repetitively. The main contributions are summarized below.

1. In Chapter 2, a general consensus tracking problem is formulated for a group of global Lipschitz continuous systems. It is assumed that the communication is fixed and connected, and the perfect identical initialization condition is satisfied as well. D-type ILC rule is proposed for the systems to achieve perfect consensus tracking. By adoption of a graph dependent matrix norm, a local convergence condition is devised at the agent level. In addition, optimal learning gain design methods are developed for both directed and undirected graphs such that the λ-norm of tracking error converges at the fastest rate.

2. In Chapter 3, we investigate the robustness of D-type learning rule against communication variations. It turns out that the controller is insensitive to iteration-varying topology. In the most general case that the learning controller is still convergent when the communication topology is uniformly strongly connected over the iteration domain.

3. In Chapter 4, PD-type learning rule is proposed to deal with imperfect initialization condition as it is difficult to ensure perfect initial conditions for all agents due to sparse information communication that only a few of the follower agents know the desired initial state. The new learning rule offers two main features. On the one hand, it can ensure controller convergence; one the other hand, the learning gain can be used to tune the final tracking performance.

4. In Chapter 5, by combining the Lyapunov analysis method and contraction-mapping analysis, we explore the applicability of P-type learning rule to several classes of local Lipschitz nonlinear systems. Several sufficient convergence conditions in terms of Lyapunov criteria are derived. In particular, the P-type learning rule can be applied to Lyapunov stable system with quadratic Lyapunov functions, exponentially stable system, system with bounded drift terms, and uniformly bounded energy bounded state system under control saturation. The results greatly complement to the existing literature.

5. In Chapter 6, composite energy function method is utilized to design adaptive learning rule to deal with local Lipschitz systems that can be modeled by linear in parameter form. With the help of a special parameterization method, the leader's trajectory can be treated as an iteration-invariant parameter that all the followers can learn from local measurements. Besides, the initial rectifying action is ap-

plied to reduce the effect of imperfect initialization condition. The method works for high-order systems as well.

6. Lagrangian systems have wide applications in practice. For example, industry robotic manipulators can be modeled by Lagrangian system. In Chapter 7, we develop a set of distributed learning rules to synchronize networked Lagrangian systems. In the controller design, we fully utilize the inherent features of Lagrangian systems, and the controller works under directed acyclic graph.

Chapter 2

Optimal Iterative Learning Control for Multi-agent Consensus Tracking

2.1 Background

The idea of using ILC for multi-agent coordination first appears in Ahn and Chen (2009), where multi-agent formation control problem is studied for a group of global Lipschitz nonlinear systems, in which the communication graph is identical to the formation structure. When the tree-like formation is considered, the perfect formation control can be achieved. In Xu et al. (2011), by incorporating with high-order internal model ILC (Liu et al., 2010), an iteratively switching formation problem is formulated and solved in the same framework. The communication graphs are supposed to be direct spanning trees as well. Liu and Jia (2012) improve the control performance in Ahn et al. (2010). The formation structure can be independent of the communication topology, and time-varying communication is assumed in Liu and Jia (2012). The convergence condition is specified at the group level by a matrix norm inequality, and the learning gain can be designed by solving a set of linear matrix inequalities (LMIs). It is not clear under what condition the set of LMIs admit a solution, and it is lack of insight how the communication topologies relate to the convergence condition. In Meng and Jia (2012), the idea of terminal ILC (Xu et al., 1999) is brought into consensus problem. A finite-time consensus problem is formulated for discrete-time linear systems in ILC framework. It is shown that all the agents reach consensus at the terminal time as iteration number goes to infinity. In Meng et al. (2012), the authors extend the terminal consensus problem in their previous work to track a time-varying reference trajectory over the entire finite-time interval. A unified ILC algorithm is developed for both discrete-time and continuous-time linear agents. Necessary and sufficient conditions in the form of spectral radius are derived to ensure the convergence properties. Shi et al. (2014) develop a learning controller for second-order multi-agent systems to perform formation control by using the similar approach.

In this chapter, we study the consensus tracking problem for a group of time-varying nonlinear dynamic agents, where the nonlinear terms satisfy the global Lipschitz continuous condition. The communication graph is assumed to be fixed. In comparison with the current literature, the main challenges and contributions are summarized below: (1) in Meng et al. (2012), the convergence condition for continuous-time agents is derived based on the result of 2-dimensional system theory (Chow and Fang, 1998), which is only valid for linear systems. By adoption of a graph dependent matrix norm and λ-norm analysis, we are able to obtain the results for global Lipschitz nonlinear systems; (2) in Liu and Jia (2012), the convergence condition is specified at the group level in the form of a matrix norm inequality, and learning gain is designed by solving a set of LMIs. Nevertheless, owing to the graph dependent matrix norm, the convergence condition is expressed at the individual agent level in the form of spectral radius inequalities in our work, which are related to the eigenvalues associated with the communication graph. It shows that these eigenvalues play crucial roles in the convergence condition. In addition, the results are less conservative than the matrix norm inequality since the spectral radius of a matrix is less or equal to its matrix norm; (3) by using the graph dependent matrix norm and λ-norm analysis, the learning controller design can be extended to heterogeneous systems; (4) the obtained convergence condition motivates us to consider optimal learning gain designs which can impose the tightest bounding functions for the actual tracking errors.

The rest of this chapter is organized as follows. In Section 2.2, notations and some useful results are introduced. Next, the consensus tracking problem for heterogeneous agents is formulated. Then, learning control laws are developed in Section 2.3, for both homogeneous and heterogeneous agents. Next, optimal learning design methods are proposed in Section 2.4, where optimal designs for undirected and directed graphs are explored respectively. Then, an illustrative example for heterogeneous agents under fixed directed graph is given in Section 2.5 to demonstrate the efficacy of the proposed algorithms. Finally, we conclude this chapter in Section 2.6.

2.2 Preliminaries and Problem Description

2.2.1 Preliminaries

The set of real numbers is denoted by $\mathbb{R}$, and the set of complex numbers is denoted by $\mathbb{Z}$. The set of integers is denoted by $\mathbb{N}$, and $i \in \mathbb{N}_{\geq 0}$ is the number of iteration. For any $z \in \mathbb{Z}$, $\Re(z)$ denotes its real part. For a given vector $\mathbf{x} = [x_1, x_2, \cdots, x_n]^T \in \mathbb{R}^n$, $|\mathbf{x}|$ denotes any l_p vector norm, where $1 \leq p \leq \infty$. In particular, $|\mathbf{x}|_1 = \sum_{k=1}^{n} |x_k|$, $|\mathbf{x}|_2 = \sqrt{\mathbf{x}^T \mathbf{x}}$, and $|\mathbf{x}|_\infty = \max_{k=1,\ldots,n} |x_k|$. For any matrix $A \in \mathbb{R}^{n\times n}$, $|A|$ is the induced matrix norm. $\rho(A)$ is its spectral radius. Moreover, $\otimes$ denotes the Kronecker product, and I_m is the $m \times m$ identity matrix.

Let $\mathscr{C}^m[0,T]$ denote a set consisting of all functions whose mth derivatives are continuous on the finite-time interval $[0,T]$. For any function $\mathbf{f}(\cdot) \in \mathscr{C}[0,T]$, the supremum norm is defined as $\|\mathbf{f}\| = \sup_{t\in[0,T]} |\mathbf{f}(t)|$. Let λ be a positive constant, the time weighted norm (λ-norm) is defined as $\|\mathbf{f}\|_\lambda = \sup_{t\in[0,T]} e^{-\lambda t}|\mathbf{f}(t)|$.

Graph theory (Biggs, 1994) is an instrumental tool to describe the communication topology among agents in the multi-agent systems, the basic terminologies and some properties of algebraic graph theory are revisited in Appendix A. Please go through Appendix A as the vertex set $\mathscr{V}$ represents the agent index and the edge set $\mathscr{E}$ describes the information flow among agents.

For simplicity, 0-1 weighting is adopted in the graph adjacency matrix $\mathscr{A}$. However, any positive weighted adjacency matrix preserves the convergence results. The strength of the weights can be interpreted as the reliability of information in the communication channels. In addition, positive weights can represent the collaboration among agents. Whereas, negative weights can represent the competition among agents. For example, Altafini (2013) shows that the consensus can be reached on signed networks but the consensus values have opposite signs. If the controller designer has the freedom to select the weightings in the adjacency matrix, Xiao and Boyd (2004) demonstrate that some of the edges may take negative weights in order to achieve the fastest convergence rate in linear average algorithm. Although interesting, negative weighting is outside the scope of this thesis.

The following propositions and lemma lay the foundations for the convergence analysis in the main results.

Proposition 2.1 *For any given matrix $M \in \mathbb{R}^{n\times n}$ satisfying $\rho(M) < 1$, there exists at least one matrix norm $|\cdot|_S$ such that $\lim_{k\to\infty} (|M|_S)^k = 0$.*

Proposition 2.1 is an extension of Lemma 5.6.10 in Horn and Johnson (1985). The proof is given in Appendix B.1 as the idea in the proof will be used to prove Theorem 2.1 and illustrate the graph dependent matrix norm.

Proposition 2.2 *(Horn and Johnson, 1985, pp. 297) For any matrix norm $|\cdot|_S$, there exists at least one compatible vector norm $|\cdot|_{\mathbf{s}}$, and for any $M \in \mathbb{R}^{n\times n}$ and $\mathbf{x} \in \mathbb{R}^n$, $|M\mathbf{x}|_{\mathbf{s}} \le |M|_S|\mathbf{x}|_{\mathbf{s}}$.*

The following Proposition 2.3, 2.4, and Lemma 2.1 will be utilized in the optimal learning gain designs.

Proposition 2.3 *(Xu and Tan, 2002b) Denoting the compact set $\mathscr{I} = [\alpha_1, \alpha_2]$, where $0 < \alpha_1 < \alpha_2 < +\infty$, the index*

$$J = \min_{\gamma\in\mathbb{R}} \max_{d\in\mathscr{I}} |1 - d\gamma|$$

reaches its minimum value $\frac{\alpha_2-\alpha_1}{\alpha_2+\alpha_1}$ when $\gamma^* = \frac{2}{\alpha_2+\alpha_1}$.

Proposition 2.4 (maximum modulus theorem) *(Zhou and Doyle, 1998) Let $f(z)$ be a continuous complex-value function defined on a compact set $\mathscr{Z}$, and analytic on the interior of $\mathscr{Z}$, then $|f(z)|$ cannot attain the maximum in the interior of $\mathscr{Z}$ unless $f(z)$ is a constant.*

By using Proposition 2.4, Lemma 2.1 is proven in Appendix B.2.

Lemma 2.1 *When $\gamma^* = \alpha_1/\alpha_2^2$, the following min-max problem reaches its optimal value*

$$\min_{\gamma \in \mathbb{R}} \max_{\alpha_1 < a < \sqrt{a^2+b^2} < \alpha_2} |1 - \gamma(a + jb)| = \frac{\sqrt{\alpha_2^2 - \alpha_1^2}}{\alpha_2}.$$

2.2.2 Problem Description

Consider a group of N heterogeneous time-varying dynamic agents who work in a repeatable control environment. Their interaction topology is depicted by graph $\mathscr{G} = (\mathscr{V}, \mathscr{E}, \mathscr{A})$, which is iteration-invariant. At the ith iteration, the dynamics of the jth agent take the following form:

$$\begin{cases} \dot{\mathbf{x}}_{i,j}(t) = \mathbf{f}_j(t, \mathbf{x}_{i,j}(t)) + B_j(t)\mathbf{u}_{i,j}(t) \\ \mathbf{y}_{i,j}(t) = C_j(t)\mathbf{x}_{i,j}(t) \end{cases}, \forall t \in [0,T],\ \forall j \in \mathscr{V}, \tag{2.1}$$

with initial condition $\mathbf{x}_{i,j}(0)$. Here $\mathbf{x}_{i,j}(t) \in \mathbb{R}^{n_j}$ is the state vector, $\mathbf{y}_{i,j}(t) \in \mathbb{R}^m$ is the output vector, $\mathbf{u}_{i,j}(t) \in \mathbb{R}^{p_j}$ is the control input. For any $j = 1, 2, \ldots, N$, the unknown nonlinear function $\mathbf{f}_j(\cdot,\cdot)$ satisfies the global Lipschitz continuous condition with respect to $\mathbf{x}$ uniformly in $t, \forall t \in [0,T]$. In addition, the time-varying matrices $B_j(t)$ and $C_j(t)$ satisfy that $B_j(t) \in \mathscr{C}^1[0,T]$ and $C_j(t) \in \mathscr{C}^1[0,T]$.

The desired consensus tracking trajectory is denoted by $\mathbf{y}_d(t) \in \mathscr{C}^1[0,T]$. Meanwhile, the state of each agent is not measurable. The only information available is the output signal of each agent.

Instead of a traditional tracking problem in ILC, in which each agent should know the desired trajectory, $\mathbf{y}_d(t)$ is only accessible to a subset of agents. We can think of the desired trajectory as a (virtual) leader, and index it by vertex 0 in the graph representation. Thus, the complete information flow can be described by another graph $\bar{\mathscr{G}} = (\mathscr{V} \cup \{0\}, \bar{\mathscr{E}}, \bar{\mathscr{A}})$, where $\bar{\mathscr{E}}$ is the edge set and $\bar{\mathscr{A}}$ is the weighted adjacency matrix of $\bar{\mathscr{G}}$.

Let $\xi_{i,j}(t)$ denote the distributed information measured or received by the jth agent at the ith iteration. More specifically,

$$\xi_{i,j}(t) = \sum_{k \in \mathscr{N}_j} a_{j,k}(\mathbf{y}_{i,k}(t) - \mathbf{y}_{i,j}(t)) + d_j(\mathbf{y}_d(t) - \mathbf{y}_{i,j}(t)), \tag{2.2}$$

where $a_{j,k}$ is the (j,k)th entry in the adjacency matrix $\mathscr{A}$, $\mathscr{N}_j$ is the neighborhood set of the jth agent, $\mathbf{y}_{i,j}(t)$ is the output of the jth agent at the ith iteration, $d_j = 1$ if agent j can access the desired trajectory, i.e., there is an edge from the virtual leader to the jth agent or $(0, j) \in \bar{\mathscr{E}}$, and $d_j = 0$ otherwise. The tracking error is defined as $\mathbf{e}_{i,j}(t) \triangleq \mathbf{y}_d(t) - \mathbf{y}_{i,j}(t)$.

The control objective is to design an appropriate iterative learning law such that the output from each agent converges to the desired trajectory $\mathbf{y}_d(t)$ when only some of the agents know the desired trajectory.

To simplify the analysis, the following assumptions are used.

Assumption 2.1 *For any $j = 1, 2, \ldots, N$, the unknown nonlinear term $\mathbf{f}_j(t, \mathbf{x})$ satisfies*

$$|\mathbf{f}_j(t, \mathbf{z}_1) - \mathbf{f}_j(t, \mathbf{z}_2)| \leq l_j |\mathbf{z}_1 - \mathbf{z}_2|, \text{ for any } \mathbf{z}_1, \mathbf{z}_2 \in \mathbb{R}^{n_j},$$

where l_j is a positive constant.

Remark 2.1 *In the existing literature, contraction-mapping (CM) based ILC is only applicable to global Lipschitz systems. Extension to local Lipschitz systems remains open. Two possible research directions are available. If the nonlinear terms can be linearly parameterized, composite energy function (CEF) based ILC (Xu and Tan, 2003) can be applied to overcome the global Lipschitz assumption. The other method makes use of the stability properties of system dynamics. By combining Lyapunov and CM analysis methods, it is possible to extend CM based ILC to certain classes of local Lipschitz continuous systems. This kind of methodology will be explored in Chapter 5.*

Assumption 2.2 *$C_j(t)B_j(t)$ is of full row rank for all $t \in [0,T]$.*

Remark 2.2 *The requirement that $C_j(t)B_j(t)$ is of full row rank for any $j = 1,2,\ldots,N$ and any $t \in [0,T]$ can be relaxed if using the higher order derivatives of $\xi_{i,j}(t)$ (if they do exist) in the learning updating law. The proof technique will be very similar. If the higher order derivatives do not exist, some smooth approximations of these higher order derivatives can be applied.*

Assumption 2.3 *The communication graph $\bar{\mathscr{G}}$ contains a spanning tree with the (virtual) leader being the root.*

Remark 2.3 *Assumption 2.3 is a necessary communication requirement for the solvability of the consensus tracking problem. If there is an isolated agent, it is impossible for that agent to follow the leader's trajectory as it does not even know the control objective. It is noted that the original communication graph $\mathscr{G}$ does not necessarily contain a spanning tree. By selecting a (virtue) leader and its communication carefully, under such a situation, the proposed updating law can still work.*

Furthermore, the following identical initialization condition (i.i.c.) is needed.

Assumption 2.4 *The systems are reset to the same initial state after each execution, and $\mathbf{e}_{i,j}(0) = 0$ for any $j = 1,2,\ldots,N$, $i \in \mathbb{N}_{\geq 0}$.*

Remark 2.4 *The i.i.c. is a standard assumption in ILC design to ensure the perfect tracking performance. It is possible to remove this condition with a sacrifice in tracking performance, but they require either extra system information or additional control mechanisms, for instance, the initial state learning rule (Chen et al., 1999) and initial rectifying action (Sun and Wang, 2002). Note that without perfect initial condition, perfect tracking can never be achieved. More discussions on various initial conditions in* *the learning context can be found in Park et al. (1999); Xu and Yan (2005); Chi et al. (2008) and references therein. It is highlighted that only the output of each agent is required to start from the same initial value as $\mathbf{y}_d(0)$. For example, in many applications, the state of the system includes the position and velocity where the output is just the velocity information. Under such a situation, it is very natural to assume that the output has zero initial velocity as the desired trajectory. The initial condition problem will be further explored in Chapters 4 and 6.*

2.3 Main Results

In the consensus literature, consensus problem is usually studied for a group of identical agents. Whereas, the problem formulation presented in systems (2.1) is very general, in which all the parameters are agent dependent.

For simplicity, the learning law is first designed for multi-agent systems with identical agents. Then the results will be extended to heterogeneous systems (2.1).

2.3.1 Controller Design for Homogeneous Agents

Assume that in (2.1), each agent has identical dynamics, that is, $\mathbf{f}_j(t,\mathbf{x}) = \mathbf{f}(t,\mathbf{x})$, $C_j(t) = C(t)$, and $B_j(t) = B(t)$ for all $j = 1, 2, \ldots, N$.

The following D-type updating law is used to solve the consensus tracking problem,

$$\mathbf{u}_{i+1,j}(t) = \mathbf{u}_{i,j}(t) + \Gamma(t)\dot{\xi}_{i,j}(t),\ \mathbf{u}_{0,j}(t) \equiv 0, \tag{2.3}$$

where $\Gamma(t) \in \mathscr{C}^1[0,T]$ is a time-varying learning gain matrix to be designed.

Remark 2.5 *The updating law (2.3) sets zero initial condition for $\mathbf{u}_{0,j}(t)$ for simplicity. Some feedback law can be used to construct $\mathbf{u}_{0,j}(t)$ such that the systems are stable in* *the time domain, which may be helpful for the transient performance in the learning process.*

The distributed measurement in (2.2) can be rewritten in terms of the tracking errors as

$$\xi_{i,j}(t) = \sum_{k \in N_j} a_{j,k}(\mathbf{e}_{i,j}(t) - \mathbf{e}_{i,k}(t)) + d_j \mathbf{e}_{i,j}(t). \tag{2.4}$$

Define three column stack vectors in the ith iteration $\mathbf{x}_i(t) = [\mathbf{x}_{i,1}(t)^T, \mathbf{x}_{i,2}(t)^T, \ldots, \mathbf{x}_{i,N}(t)^T]^T$, $\mathbf{e}_i(t) = [\mathbf{e}_{i,1}(t)^T, \mathbf{e}_{i,2}(t)^T, \ldots, \mathbf{e}_{i,N}(t)^T]^T$, and $\xi_i(t) = [\xi_{i,1}(t)^T, \xi_{i,2}(t)^T, \ldots, \xi_{i,N}(t)^T]^T$. Consequently, (2.4) can be written in a compact form

$$\xi_i(t) = ((L + D) \otimes I_m)\mathbf{e}_i(t), \tag{2.5}$$

where L is the Laplacian matrix of graph $\mathscr{G}$, and $D \triangleq \mathrm{diag}(d_1, d_2, \ldots, d_N)$.

By using (2.5), the updating law (2.3) can be rewritten in terms of the tracking errors.

$$\mathbf{u}_{i+1}(t) = \mathbf{u}_i(t) + ((L + D) \otimes \Gamma(t))\dot{\mathbf{e}}_i(t). \tag{2.6}$$

For convenience, we define $\lambda_j, j = 1, 2, \ldots, N$ as the jth eigenvalue of $L + D$.

The following theorem summarizes the convergence properties of the consensus algorithms (2.6).

Theorem 2.1 *Assume that Assumptions 2.1 – 2.4 hold for the time-varying nonlinear systems (2.1) with the systems' parameters being identical. If the learning gain matrix* $\Gamma(t)$ *satisfies the following condition,*

$$\max_{j=1,2..N} \max_{t \in [0,\, T]} \rho\left(I_m - \lambda_j \cdot C(t) \cdot B(t) \cdot \Gamma(t)\right) \leq \rho < 1, \tag{2.7}$$

for some $\rho \in (0,1)$*, then there exists a positive constant* λ *such that*

$$\|\mathbf{e}_{i+1}\|_\lambda \leq \rho \|\mathbf{e}_i\|_\lambda, \forall j = 1, 2, \ldots, N, \tag{2.8}$$

which indicates that $\lim_{i\to\infty} \mathbf{y}_{i,j}(t) = \mathbf{y}_d(t)$ *for all* $t \in [0,\ T],\ j = 1,2,\ldots,N.$

Proof: The tracking error of the jth agent between two consecutive iterations can be expressed as

$$\begin{aligned} \mathbf{e}_{i+1,j}(t) &= \mathbf{y}_d(t) - \mathbf{y}_{i+1,j}(t) \\ &= \mathbf{e}_{i,j}(t) - (\mathbf{y}_{i+1,j}(t) - \mathbf{y}_{i,j}(t)), \end{aligned}$$

which can be written as the following compact form,

$$\mathbf{e}_{i+1}(t) = \mathbf{e}_i(t) - (I_N \otimes C(t))(\mathbf{x}_{i+1}(t) - \mathbf{x}_i(t)). \tag{2.9}$$

The state difference $\mathbf{x}_{i+1}(t) - \mathbf{x}_i(t)$ can be calculated by integrating the system dynamics (2.1) along the time domain,

$$\begin{aligned} &\mathbf{x}_{i+1}(t) - \mathbf{x}_i(t) \\ &= \mathbf{x}_{i+1}(0) - \mathbf{x}_i(0) + \int_0^t \left(\bar{\mathbf{f}}(\tau, \mathbf{x}_{i+1}) - \bar{\mathbf{f}}(\tau, \mathbf{x}_i) + (I_N \otimes B(t))(\mathbf{u}_{i+1}(\tau) - \mathbf{u}_i(\tau))\right) d\tau \end{aligned} \tag{2.10}$$

where $\bar{\mathbf{f}}(t, \mathbf{x}_i) \triangleq [\mathbf{f}(t, \mathbf{x}_{i,1})^T, \mathbf{f}(t, \mathbf{x}_{i,2})^T, ..., \mathbf{f}(t, \mathbf{x}_{i,N})^T]^T$.

According to Assumption 2.4, and using the updating law (2.6), it yields

$$\begin{aligned} &\mathbf{x}_{i+1}(t) - \mathbf{x}_i(t) \\ &= \int_0^t \left(\bar{\mathbf{f}}(\tau, \mathbf{x}_{i+1}) - \bar{\mathbf{f}}(\tau, \mathbf{x}_i) + (L+D) \otimes (B(\tau)\Gamma(\tau))\dot{\mathbf{e}}_i(\tau)\right) d\tau. \end{aligned} \tag{2.11}$$

Apply integration by parts to the last term in (2.11), noting Assumption 2.4, it has

$$\begin{aligned} &\int_0^t (L+D) \otimes (B(\tau)\Gamma(\tau))\dot{\mathbf{e}}_i(\tau)\, d\tau \\ &= (L+D) \otimes (B(t)\Gamma(t))\mathbf{e}_i(t) + \int_0^t (L+D) \otimes \left(\frac{d}{d\tau} B(\tau)\Gamma(\tau)\right) \mathbf{e}_i(\tau)\, d\tau. \end{aligned} \tag{2.12}$$

Substituting (2.12) into (2.11), we can obtain

$$\begin{aligned}\mathbf{x}_{i+1}(t)-\mathbf{x}_i(t) &= (L+D)\otimes B(t)\Gamma(t)\mathbf{e}_i(t)+\int_0^t [\bar{\mathbf{f}}(\tau,\mathbf{x}_{i+1})-\bar{\mathbf{f}}(\tau,\mathbf{x}_i)]\,d\tau \\ &\quad +\int_0^t \left((L+D)\otimes\left(\frac{d}{d\tau}B(\tau)\Gamma(\tau)\right)\mathbf{e}_i(\tau)\right)\,d\tau. \end{aligned}\tag{2.13}$$

Then, substituting (2.13) to (2.9) yields

$$\begin{aligned}\mathbf{e}_{i+1}(t) &= (I_{mN}-(L+D)\otimes C(t)B(t)\Gamma(t))\mathbf{e}_i(t)-(I_N\otimes C(t))\left(\int_0^t [\bar{\mathbf{f}}(\tau,\mathbf{x}_{i+1})-\bar{\mathbf{f}}(\tau,\mathbf{x}_i)]\,d\tau+\right. \\ &\left.\int_0^t \left((L+D)\otimes\left(\frac{d}{d\tau}B(\tau)\Gamma(\tau)\right)\mathbf{e}_i(\tau)\right)\,d\tau\right). \end{aligned}\tag{2.14}$$

For simple presentation, the following constants are used in the sequel.

$$\begin{aligned} b_1 &\triangleq \|I_N\otimes C(t)\|, \\ b_2 &\triangleq \left\|(L+D)\otimes\left(\frac{d}{dt}B(t)\Gamma(t)\right)\right\|, \\ b_3 &\triangleq \|(L+D)\otimes B(t)\Gamma(t)\|. \end{aligned}$$

Taking norm on both sides of (2.14), and noticing the Lipschitz condition in Assumption 2.1, we have

$$\begin{aligned}|\mathbf{e}_{i+1}(t)| &\leq \|I_{mN}-(L+D)\otimes C(t)B(t)\Gamma(t)\|\,|\mathbf{e}_i(t)| \\ &\quad +b_1\bar{k}_f\int_0^t |\mathbf{x}_{i+1}(\tau)-\mathbf{x}_i(\tau)|\,d\tau+b_1b_2\int_0^t|\mathbf{e}_i(\tau)|\,d\tau, \end{aligned}\tag{2.15}$$

where $\bar{k}_f$ is the global Lipschitz constant of $\bar{\mathbf{f}}(\cdot,\cdot)$.

Furthermore, taking λ-norm on both sides of (2.15) yields

$$\begin{aligned}\|\mathbf{e}_{i+1}\|_\lambda &\leq \|I_{mN}-(L+D)\otimes C(t)B(t)\Gamma(t)\|\,\|\mathbf{e}_i\|_\lambda \\ &\quad +b_1\bar{k}_f\|\mathbf{x}_{i+1}-\mathbf{x}_i\|_\lambda\frac{1-e^{-\lambda T}}{\lambda}+b_1b_2\|\mathbf{e}_i\|_\lambda\frac{1-e^{-\lambda T}}{\lambda}. \end{aligned}\tag{2.16}$$

To derive the convergence property of $\|\mathbf{e}_i\|_\lambda$ along the iteration axis, it suffices to explore the relation between $\|\mathbf{x}_{i+1}-\mathbf{x}_i\|_\lambda$ and $\|\mathbf{e}_i\|_\lambda$.

Taking norm on both sides of (2.13), and applying the Lipschitz condition for system nonlinearity $\bar{\mathbf{f}}(\cdot,\cdot)$, it is shown that

$$|\mathbf{x}_{i+1}(t)-\mathbf{x}_i(t)| \quad \leq \quad b_3|\mathbf{e}_i(t)|+\bar{k}_f\int_0^t |\mathbf{x}_{i+1}(\tau)-\mathbf{x}_i(\tau)|\,d\tau+b_2\int_0^t |\mathbf{e}_i(\tau)|\,d\tau \tag{2.17}$$

Next, taking λ-norm on both sides of (2.17) yields

$$\|\mathbf{x}_{i+1}-\mathbf{x}_i\|_\lambda \quad \leq \quad b_3\|\mathbf{e}_i\|_\lambda+\bar{k}_f\frac{1-e^{-\lambda T}}{\lambda}\|\mathbf{x}_{i+1}-\mathbf{x}_i\|_\lambda+b_2\frac{1-e^{-\lambda T}}{\lambda}\|\mathbf{e}_i\|_\lambda. \tag{2.18}$$

Rearrange $\|\mathbf{x}_{i+1}-\mathbf{x}_i\|_\lambda$ and $\|\mathbf{e}_i\|_\lambda$ in (2.18),

$$\|\mathbf{x}_{i+1}-\mathbf{x}_i\|_\lambda \leq (b_3+b_2\frac{1-e^{-\lambda T}}{\lambda})(1-\bar{k}_f\frac{1-e^{-\lambda T}}{\lambda})^{-1}\|\mathbf{e}_i\|_\lambda. \tag{2.19}$$

Substituting (2.19) to (2.16), $\|\mathbf{e}_{i+1}\|_\lambda$ becomes

$$\|\mathbf{e}_{i+1}\|_\lambda \quad \leq \quad \|I_{mN}-(L+D)\otimes C(t)B(t)\Gamma(t)\|\,\|\mathbf{e}_i\|_\lambda+\mathscr{O}(\lambda^{-1})\|\mathbf{e}_i\|_\lambda, \tag{2.20}$$

where

$$\mathscr{O}(\lambda^{-1}) \quad = \quad b_1\bar{k}_f(b_3+b_2\frac{1-e^{-\lambda T}}{\lambda})(1-\bar{k}_f\frac{1-e^{-\lambda T}}{\lambda})^{-1}\frac{1-e^{-\lambda T}}{\lambda}+b_1b_2\frac{1-e^{-\lambda T}}{\lambda}.$$

As $\mathscr{O}(\lambda^{-1})$ is in the same order with λ^{-1}, hence, it can be made negligibly small by choosing a sufficiently large λ. As a result if

$$\|I_{mN}-(L+D)\otimes C(t)B(t)\Gamma(t)\| \leq \rho < 1, \tag{2.21}$$

then, there exists $0<\rho<\rho_1<1$ such that $\|\mathbf{e}_{i+1}\|_\lambda \leq \rho_1\|\mathbf{e}_i\|_\lambda$.

Define $M(t) \triangleq I_{mN}-(L+D)\otimes C(t)B(t)\Gamma(t)$. Based on Proposition 2.1, it is sufficient to design a suitable $\Gamma(t)$ such that $\rho(M(t)) \leq \rho,$ for all $t\in[0,T]$, then the consensus tracking is fulfilled by (2.6). This is because when $\rho(M(t))\leq\rho$ for all $t\in[0,T]$, we can always find an appropriate matrix norm such that $\|M\|\leq\rho$.

Note that $M(t)\in\mathbb{R}^{mN\times mN}$, and the condition $\rho(M(t))\leq\rho,\forall t\in[0,T]$ is specified at the group level since it contains all the agents' dynamics and the complete communication topology. It does not make much difference in the homogeneous case. However, we will see later that it becomes more complex in the heterogeneous case. Next, let us derive the convergence condition at the agent level. For simplicity in the sequel, the time argument is dropped when no confusion arises.

Follow the concepts in Appendix A, $L+D$ can be decomposed in the following form

$$\Delta = U^*(L+D)U,$$

where Δ is an upper triangular matrix with diagonal entries being the eigenvalues of $L+D$, U is an associated unitary matrix, and * denotes the conjugate transpose.

Let the matrix norm operation in the above development be defined as below

$$|\cdot| \triangleq \left|[(QU^*)\otimes I_m]\,(\cdot)\,[(UQ^{-1})\otimes I_m]\right|,$$

where Q is a constant matrix defined in Appendix A. Based on Proposition 2.2, there always exists a corresponding vector norm which is compatible to previously defined matrix norm. Hence, all the derivations here remain valid.

Then, we have

$$\begin{aligned}
&|M| \\
&= \left|[(QU^*)\otimes I_m][I_{mN}-(L+D)\otimes CB\Gamma][(UQ^{-1})\otimes I_m]\right| \\
&= \left|I_{mN}-[(QU^*)(L+D)(UQ^{-1})]\otimes CB\Gamma\right| \\
&= \left|I_{mN}-(Q\Delta Q^{-1})\otimes CB\Gamma\right| \\
&= \left|\begin{bmatrix} I_m-\lambda_1 CB\Gamma & (\star) & (\star) & (\star) \\ 0 & I_m-\lambda_2 CB\Gamma & (\star) & (\star) \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & I_m-\lambda_N CB\Gamma \end{bmatrix}\right|,
\end{aligned} \tag{2.22}$$

where $(\star)$ can be made arbitrarily small by choosing sufficiently large α in Q.

By Assumption 2.3 the leader has a path to any follower agent, then $L+D$ is nonsingular and all the eigenvalues have positive real parts (Ren and Beard, 2008). Together with (2.22), we can conclude that if

$$\max_{j=1,2..N} \max_{t\in[0,T]} \rho\left(I_m - \lambda_j C(t)B(t)\Gamma(t)\right) \leq \rho,$$

then equation (2.21) holds, i.e., the tracking error $\mathbf{e}_i(t)$ converges to zero as i goes to infinity. ■

Remark 2.6 *The matrix norm defined in the proof depends on the communication graph $\bar{\mathscr{G}}$ as U is calculated from $L+D$, that is why we call it the graph dependent norm. Such a norm enables us to derive a simpler convergence condition, and reveals the insight of relation between communication topology and convergence property.*

The convergence condition (2.7) is specified at the agent level, and it has the same form as the traditional D-type ILC convergence condition. The influence of communication to the convergence condition is reflected through the eigenvalues λ_j, $j=1,2,\ldots,N$, which are associated with the graph $\bar{\mathscr{G}}$. Hence, the local controller design is decoupled from other agents' dynamics. This motivates us to consider the heterogeneous agents. Surprisingly, similar convergence condition applies to heterogeneous agents as well. Detailed treatment is discussed in Section 2.3.2.

Remark 2.7 *In Assumption 2.2, CB is assumed to be of full row rank. We can select $\Gamma = \gamma(CB)^T(CB(CB)^T)^{-1}$, then convergence condition becomes much simpler,*

$$\max_{j=1,2..N} |1-\gamma\lambda_j| < 1. \tag{2.23}$$

This expression motivates us to consider the optimal learning gain design in the sense that $\|\mathbf{e}_i\|_\lambda$ converges at the fastest rate, which is explored in Section 2.4.

2.3.2 Controller Design for Heterogeneous Agents

Heterogeneity is the nature of multi-agent systems. Even for the same type of agents, they may have the similar structures, but it is unlikely that they share the identical parameters. Consensus tracking problem for heterogeneous agents is more practical, but of course more challenging.

Consider the model in (2.1). Notice that unlike the homogeneous systems case, all the agent dynamics are different from each other now. As the target is for output consensus tracking, the outputs from all agents should have the same dimension. Whereas, the state dimensions may not necessarily be the same.

Now we adopt the following learning rule,

$$\mathbf{u}_{i+1,j} = \mathbf{u}_{i,j} + \Gamma_j \dot{\xi}_{i,j}, \tag{2.24}$$

where Γ_j is the agent dependent learning gain matrix to be designed. The controller (2.24) is similar to (2.3) in the homogeneous case, except that the learning gain Γ_j is agent dependent.

Writing (2.24) in a compact form, we have

$$\mathbf{u}_{i+1} = \mathbf{u}_i + \Gamma((L+D)\otimes I_m)\dot{\mathbf{e}}_i(t), \tag{2.25}$$

where $\Gamma = \text{diag}(\Gamma_1, \Gamma_2, \ldots, \Gamma_N)$. Due to the heterogeneity, the structure of (2.25) is rather different from the one in homogeneous case. Follow the similar development in the previous section, eventually, we have

$$\|\mathbf{e}_{i+1}\|_\lambda \leq \|I_{mN} - CB\Gamma((L+D)\otimes I_m)\| \, \|\mathbf{e}_i\|_\lambda + \mathscr{O}(\lambda^{-1})\|\mathbf{e}_i\|_\lambda, \tag{2.26}$$

where $B = \text{diag}(B_1, B_2, \ldots, B_N)$, and $C = \text{diag}(C_1, C_2, \ldots, C_N)$. If the agent dynamics are identical, i.e., C_j, B_j, and Γ_j are identical for different j, then (2.26) degenerates to (2.20). That means the homogeneous case is a special case of heterogeneous case. Let $M = I_{mN} - CB\Gamma((L+D)\otimes I_m)$, of course, if $\rho(M) \leq \rho < 1$, $\forall t \in [0,T]$, then the consensus tracking erorr $\mathbf{e}_i(t)$ converges to zero along the iteration axis. The result is summarized in the following corollary.

Corollary 2.1 *Assume that Assumptions 2.1 – 2.4 hold for the time-varying nonlinear systems (2.1). Under the control law (2.24), if the learning gain matrix* Γ *satisfies the following condition,*

$$\max_{t\in[0,\,T]} \rho\left(I_{mN} - CB\Gamma((L+D)\otimes I_m)\right) \leq \rho < 1,$$

for some $\rho \in (0,1)$*, then we have that* $\lim_{i\to\infty} \mathbf{y}_{i,j}(t) = \mathbf{y}_d(t)$ *for all* $t \in [0,\,T]$, $j = 1,2,\ldots,N$.

However, such a convergence condition is specified at the group level, the one at agent level is preferred. Since $CB\Gamma$ and $(L+D)\otimes I_m$ do not commute in general, the matrix norm defined in previous section cannot be directly applied. Let the learning gain $\Gamma_j = \gamma(C_jB_j)^T(C_jB_j(C_jB_j)^T)^{-1}$, then

$$M = I_{mN} - \gamma((L+D)\otimes I_m).$$

By using the matrix norm defined previously, we can get the following corollary.

Corollary 2.2 *Let the learning gains* $\Gamma_j = \gamma(C_jB_j)^T(C_jB_j(C_jB_j)^T)^{-1}$*. Under the iterative learning rule (2.24), the heterogeneous agent systems (2.1) can achieve perfect consensus tracking along the iteration axis, i.e.,* $\lim_{i\to\infty} \mathbf{y}_{i,j}(t) = \mathbf{y}_d(t)$ *for* $t \in [0,T]$*, if*

$$\max_{j=1,2,\ldots,N} |1-\gamma\lambda_j| < 1, \tag{2.27}$$

where λ_j *is an eigenvalue of* $L+D$*.*

By setting the learning gains equal to $\Gamma_j = \gamma(C_jB_j)^T(C_jB_j(C_jB_j)^T)^{-1}$, the convergence condition in the heterogeneous case renders the same condition as in the homogeneous case. This can be seen from (2.23) and (2.27).

2.4 Optimal Learning Gain Design

Most ILC controllers converge asymptotically along the iteration axis. Faster and monotonic convergence is usually desired. In the continuous-time system, the convergence analysis relies on the λ-norm, which is essentially an exponentially time weighted norm. However, it is well known that the monotonic convergence of λ-norm of error does not imply the monotonic convergence of actual error. In fact, the actual error may increase to a huge magnitude, and then gradually settles down to zero. Designing controllers that enable monotonic convergence of error is still open for continuous-time systems.

In this section, we are not trying to solve the monotonic convergence problem. The optimal learning gain is designed in the sense that the λ-norm of error converges at the fastest rate, which indeed imposes a tightest bounding function for the actual tracking error. From the time weighted norm definition, it is not difficult to see that the λ-norm of error is equivalent to the norm of error, i.e.,

$$\|\mathbf{e}_i\|_\lambda \leq \|\mathbf{e}_i\| \leq e^{\lambda T}\|\mathbf{e}_i\|_\lambda,$$

where λ is a positive constant. From the above inequality, we notice that $\|\mathbf{e}_i\|$ is bounded by a constant times $\|\mathbf{e}_i\|_\lambda$. We still do not know how exactly the error behaves, but a faster decaying $\|\mathbf{e}_i\|_\lambda$ does impose a tighter bounding function for $\|\mathbf{e}_i\|$. This is the main rational for the development here. The systems' transient responses depend on not only the controllers, but also the systems dynamics. However, the time-varying nonlinear terms $\mathbf{f}_j(t,\mathbf{x})$ are unknown. So ensuring the fastest convergence of $\|\mathbf{e}_i\|_\lambda$ is the best one can expect.

By setting $\Gamma_j = \gamma(C_jB_j)^T(C_jB_j(C_jB_j)^T)^{-1}$, it has been shown in the previous two sections that both homogeneous and heterogeneous systems render the same convergence condition. Thus, the optimal designs for the two different cases can be unified. The decaying rate of $\|\mathbf{e}_i\|_\lambda$ is determined by

$$\max_{\omega} |1-\omega\gamma|, \tag{2.28}$$

where $\omega \in \Lambda \triangleq \{\lambda_1, \lambda_2, \ldots, \lambda_N\}$, is an eigenvalue of $L+D$. Depending on the communication graph, if the graph is undirected, the eigenvalues of $L+D$ are positive real numbers. Whereas, for directed graph, the eigenvalues contain complex numbers in general, and all the eigenvalues have positive real parts. Hence, optimal learning gain designs for undirected and directed graphs have to be discussed separately.

To achieve the fastest convergence rate for $\|\mathbf{e}_i\|_\lambda$, (2.28) should be minimized by γ. Subsequently, we can formulate the optimal learning gain design problem as a *min-max* optimization problem, and the central task is to find an optimal γ^* such that the objective function given in (2.29) is minimized.

$$J = \min_{\gamma} \max_{\omega \in \Lambda} |1-\omega\gamma|. \tag{2.29}$$

Theorem 2.2 gives the optimal solution to the optimization problem (2.29) when the underlying communication graph is undirected.

Theorem 2.2 *When the communication graph among followers is undirected, i.e., $L+D$ is symmetric, the solution to the optimization problem (2.29) is*

$$\gamma^* = \frac{2}{\lambda_1+\lambda_N}, \quad J_{min} = \frac{\lambda_N-\lambda_1}{\lambda_N+\lambda_1},$$

where $\lambda_1 = \min(\Lambda)$, $\lambda_N = \max(\Lambda)$.

Theorem 2.2 is a direct application of Proposition 2.3, hence, the proof is omitted. In Theorem 2.2, the minimum J can be written as

$$J_{min} = \frac{1-\lambda_1/\lambda_N}{1+\lambda_1/\lambda_N},$$

where $\frac{\lambda_1}{\lambda_N}$ is usually called the eigenratio (Altafini, 2013; Duan and Chen, 2012) in consensus (synchronization) literature. So both the smallest and largest eigenvalues of $L+D$ together determine the optimal convergence rate. If the eigenratio is close to 1, we can get a faster convergence rate. This is a unique feature of ILC based consensus algorithm.

When the communication topology is directed, the eigenvalues of $L+D$ can be complex. It is difficult to derive a general solution for (2.29) because Λ is a set of complex numbers. To find a suboptimal solution, define a compact region Ω, which contains all the eigenvalues of $L+D$,

$$\Omega = \{z \in \mathbb{Z} \,|\, \Re(z) \geq \min_j(\Re(\lambda_j)) = \alpha_1,\ |z| \leq \max_j(|\lambda_j|) = \alpha_2\}.$$

Obviously, $\Lambda \subset \Omega$. Hence, the solution for

$$J = \min_{\gamma} \max_{\omega \in \Omega} |1 - \omega\gamma| \tag{2.30}$$

is a suboptimal solution for (2.29). Now it is straightforward to extend the result to directed graph by using Lemma 2.1.

Theorem 2.3 *When the communication graph among followers is directed, i.e., $L+D$ is asymmetric, a suboptimal solution to the problem (2.29) is*

$$\gamma^* = \frac{\alpha_1}{\alpha_2^2},\ J_{min} = \frac{\sqrt{\alpha_2^2 - \alpha_1^2}}{\alpha_2},$$

where $\alpha_1 = \min_{j=1,2,\ldots N}(\Re(\lambda_j))$, $\alpha_2 = \max_{j=1,2,\ldots N}(|\lambda_j|)$.

Remark 2.8 *To find out the suboptimal solution, we purposely enlarge the search region from Λ to Ω, where $\Lambda \subset \Omega$. Hence, the suboptimal solution may be conservative. However, suboptimal solution has two advantages. On the one hand, the method can be applied when the eigenvalues are not exactly known, but the region where they are* *located is known. On the other hand, the suboptimal solution is robust against small variations in the communication graph. When the exact eigenvalues are known, numerical methods should be adopted to minimize the objective function (2.29), which renders a global optimal solution.*

Remark 2.9 *For the optimal learning gain designs, one needs the complete information of the communication topology. This is the trade-off between systems information availability and performance. When the number of agents is huge, obtaining the eigenvalues of* $L+D$ *is computationally difficult. The analytical bounds of eigenvalues can be calculated from the results in Zhang (2011). Alternatively, Gershgorin disk theorem (Horn and Johnson, 1985) can be applied to estimate the eigenvalue region. When the detailed information of the graph is unknown, we can simply set* $0 < \gamma < 1/\max_j(l_{jj}+d_j)$*. If the leader agent has a path to any follower agent, this ensures that* $\rho(M) < 1$ *and consensus tracking can be achieved along the iteration axis.*

2.5 Illustrative Example

To illustrate the efficacy of the proposed consensus schemes, consider a network consisting of four heterogeneous follower agents. The agent models are governed by

$$\begin{cases} \dot{\mathbf{x}}_{i,1} = \begin{bmatrix} 0.5\cos(x_{i,1_1}) + 0.2x_{i,1_2} \\ -0.1\sin(x_{i,1_2}) \end{bmatrix} + \begin{bmatrix} 1 & 1.5 \\ 1 & -1 \end{bmatrix} \mathbf{u}_{i,1}, \\ y_{i,1} = \begin{bmatrix} 1 & 1 \end{bmatrix} \mathbf{x}_{i,1}; \end{cases}$$

$$\begin{cases} \dot{\mathbf{x}}_{i,2} = \begin{bmatrix} -0.1x_{i,2_1} + 0.2x_{i,2_2} \\ \sin(t)x_{i,2_1} \end{bmatrix} + \begin{bmatrix} 1 & 2 \\ 3 & 4 \end{bmatrix} \mathbf{u}_{i,2}, \\ y_{i,2} = \begin{bmatrix} 2 & 3 \end{bmatrix} \mathbf{x}_{i,2}; \end{cases}$$

$$\begin{cases} \dot{\mathbf{x}}_{i,3} = \begin{bmatrix} 0.2\sin(x_{i,3_1}) \\ -0.2\sin(x_{i,3_2}) \\ -0.4\cos(x_{i,3_3}) \end{bmatrix} + \begin{bmatrix} 1 & 2 \\ 3 & 4 \\ 0.5 & 0.6 \end{bmatrix} \mathbf{u}_{i,3}, \\ y_{i,3} = \begin{bmatrix} 1 & 1 & 0.5 \end{bmatrix} \mathbf{x}_{i,3}; \end{cases}$$

$$\begin{cases} \dot{\mathbf{x}}_{i,4} = \begin{bmatrix} -0.2x_{i,4_1} - 0.1x_{i,4_2} \\ \sin(x_{i,4_2}) \\ -0.3x_{i,4_3} \end{bmatrix} + \begin{bmatrix} 1 & -1 \\ 0.5 & 1 \\ 2 & 1 \end{bmatrix} \mathbf{u}_{i,4}, \\ y_{i,4} = \begin{bmatrix} 0.5 & 0.7 & 0.3 \end{bmatrix} \mathbf{x}_{i,4}. \end{cases}$$

The desired reference trajectory is

$$y_d = t + 2\sin(t),\ t \in [0,5]. \tag{2.31}$$

The information exchange among followers is assumed to be fixed and directed.

Figure. 2.1 shows the information flow among agents. The virtual leader is labeled by vertex 0 in the communication graph, and it has edges (dash arrows) to agents 1 and 4. The communication among followers is depicted by solid arrows. Notice that the

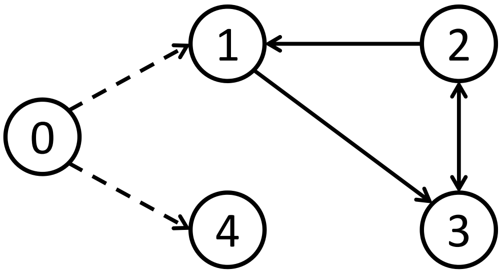

Figure 2.1: Communication topology among agents in the network.

communication graph among followers is not connected. However, the communication graph including the leader contains a spanning tree with the leader being the root. We adopt $0-1$ weighting, thus, the Laplacian for follower agents is

$$L = \begin{bmatrix} 1 & -1 & 0 & 0 \\ 0 & 1 & -1 & 0 \\ -1 & -1 & 2 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix},$$

and $D = \mathrm{diag}(1,0,0,1)$. The eigenvalues of $L+D$ are $\{0.16, 1.00, 2.42 \pm j0.61\}$ (j here denotes $\sqrt{-1}$). Since the eigenvalues are exactly known, we adopt a linear search algorithm to find out the optimal γ^* that minimizes objective function (2.29). It turns out that $\gamma^* = 0.73$. Let the learning gains

$$\Gamma_j = \gamma^*(C_jB_j)^T(C_jB_j(C_jB_j)^T)^{-1},$$

where $j = 1,2,3,4$. Check the convergence condition (2.27), we can obtain that

$$\max_{j=1,2,3,4} |1-\lambda_j\gamma^*| = 0.88 < 1.$$

Therefore, the convergence condition in Corollary 2.2 is satisfied, and the consensus tracking is achievable by the ILC rule (2.24).

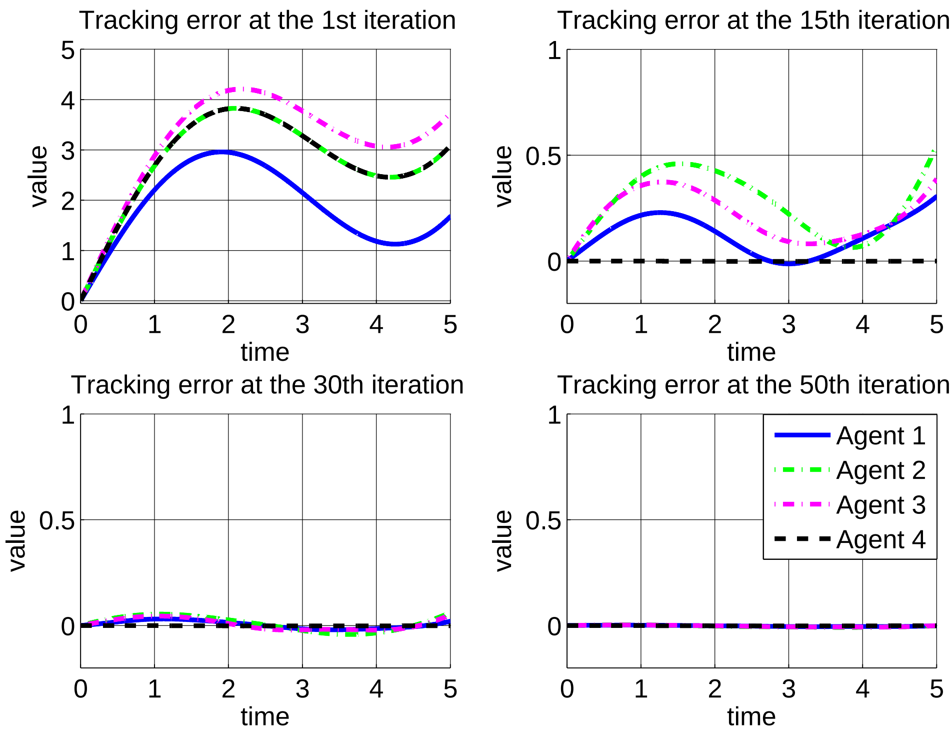

Figure 2.2: Tracking errors of all agents at different iterations.

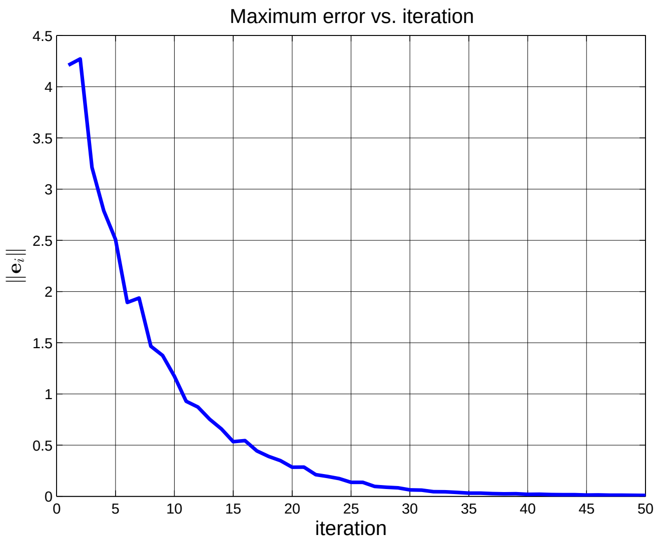

Figure 2.3: Maximum tracking error vs. iteration number.

In the simulation example, the identical initial condition is assumed, i.e., $\mathbf{e}_i(0) = 0$, and all the agents are reset to the same initial position after one iteration. The control signals at the zeroth iteration are set to zero, i.e., $\mathbf{u}_{0,j} = 0$, for all agents. Figure. 2.2 shows the agents' output tracking errors $y_d(t) - y_{i,j}(t)$ at the 1st, 15th, 30th, and 50th iterations. At the 1st iteration, the trajectories of followers have very large deviations from the desired one. As it can be seen from Figure. 2.2, the tracking errors are gradually reduced by the learning controllers, and they are almost eliminated at the 50th iteration. Figure. 2.3 shows the maximum error convergence profile versus iteration number. As the iteration number increases, all agents' outputs asymptotically converge to the desired trajectory (2.31).

2.6 Conclusion

In this chapter, a consensus tracking problem is formulated for a group of global Lipschitz nonlinear systems. A distributed D-type ILC control law is studied for the consensus tracking problem for both homogeneous and heterogeneous systems. By adoption of a graph dependent matrix norm, the convergence condition is specified at the agent level, which makes the learning gain design decoupled from other agents' dynamics. In addition, optimal learning gain design methods are developed for both undirected and directed graphs, in the sense that the λ-norm of the tracking error decays at the fastest rate, which imposes a tightest bounding function for the actual tracking error. A consensus tracking example for heterogeneous agent systems under directed graph is given to demonstrate the effectiveness of the developed design methods.

Chapter 3

Iterative Learning Control for Multi-agent Coordination Under Iteration-varying Graph

3.1 Background

Multi-agent coordination and control problems are usually studied under fixed communication topology. However, the fixed communication is restrictive and difficult to be maintained by the multi-agent systems. The switching communication is more general and has profound implications on implementation issues. If the developed controller works under switching graph, that means the controller is more robust to communication variations such as link failure and creation. Thus it is important to study the switching communication topology and its impact on the convergence results.

Consensus problem under switching graph has been investigated by many researchers (Olfati-Saber and Murray, 2004; Hatano and Mesbahi, 2005; Moreau, 2005; Cao et al., 2005; Wu, 2006; Hong et al., 2006; Zhang and Tian, 2009). An excellent survey paper on communication assumptions and convergence results is reported by Fang and Antsaklis (2006). Many interesting results are explored in the literature, for example, the consensus results under uniformly connected union graph or random graph. However, it is still unclear how the switching communication affects the stability and convergence of ILC based controllers for multi-agent coordination. In the existing consensus tracking by ILC works Ahn and Chen (2009); Xu et al. (2011); Yang et al. (2012); Yang and Xu (2012); Meng et al. (2012); Liu and Jia (2012); Li and Li (2013), the communication topology is assumed to be fixed, except for Liu and Jia (2012) in which the communication graph is time-varying but connected at every time instance. This chapter focuses on analyzing the stability and convergence properties of the learning controller under iteration-varying communication topology. First, the convergence property is derived for fixed strongly connected graph. The analyzing method will be used to derive more general results later. Next, the results are developed for iteration-varying graph which is strongly connected in each iteration. Lastly, the results are generalized to the uniformly strongly connected graph along the iteration axis. Very recently we noticed some publications on switching graph by applying ILC. The authors in Meng et al. (2013a,b, 2014) studied the similar problem and obtained some meaningful results. The fundamental difference is that they adopted 2D system theory approach to analyze the controller convergence for discrete-time system models. Whereas, we are applying λ-norm methods for continuous-time models.

This chapter is organized as follows. The consensus tracking problem under switching communication is formulated in Section 3.2. Next, the convergence results under various communication assumptions are developed in Section 3.3. To verify the results, a numerical example is presented in Section 3.4. Lastly, Section 3.5 draws the conclusion.

3.2 Problem Description

Consider a group of N dynamic agents, and the jth agent is governed by the following nonlinear model,

$$\begin{cases} \dot{\mathbf{x}}_{i,j}(t) = \mathbf{f}(\mathbf{x}_{i,j}(t)) + B\mathbf{u}_{i,j}(t) \\ \mathbf{y}_{i,j}(t) = C\mathbf{x}_{i,j}(t) \end{cases} \quad \forall j \in \mathscr{V}, \tag{3.1}$$

where i denotes the iteration number, $\mathbf{x}_{i,j} \in \mathbb{R}^n$ is the state vector, $\mathbf{y}_{i,j} \in \mathbb{R}^m$ is the output vector, $\mathbf{u}_{i,j} \in \mathbb{R}^p$ is the control input, $\mathbf{f}(\mathbf{x}_{i,j})$ is a global Lipschitz nonlinear function of $\mathbf{x}_{i,j}$, and B, C are constant matrices of compatible dimensions. In particular, CB is of full column rank, $\mathbf{f}(\cdot)$ is unknown and satisfies

$$|\mathbf{f}(\mathbf{z}_1) - \mathbf{f}(\mathbf{z}_2)| \leq L_f|\mathbf{z}_1 - \mathbf{z}_2|, \text{ for any } \mathbf{z}_1, \mathbf{z}_2 \in \mathbb{R}^n,$$

where L_f is an unknown Lipschitz constant.

For simplicity, the time argument, t, is dropped when no confusion arises.

The desired trajectory $\mathbf{y}_d(t)$ is defined on a finite-time interval $[0, T]$, which is generated by the following dynamics,

$$\begin{cases} \dot{\mathbf{x}}_d = \mathbf{f}(\mathbf{x}_d) + B\mathbf{u}_d, \\ \mathbf{y}_d = C\mathbf{x}_d. \end{cases} \tag{3.2}$$

The communication topology among the agents is described by an iteration-varying graph $\mathscr{G}(i) = (\mathscr{V}, \mathscr{E}(i), \mathscr{A}(i))$. The desired trajectory can be treated as a virtual leader, and denote it as vertex 0 in the graph representation. Assume that only a few of the agents in the network know the desired trajectory. Therefore vertex 0 has directed edges to those follower agents who have the knowledge of $\mathbf{y}_d$ in the graph representation. Together with $\mathscr{G}(i)$, the complete information flow among the followers and the leader can be depicted by a new graph $\bar{\mathscr{G}}(i) = (\mathscr{V} \cup \{0\}, \bar{\mathscr{E}}(i), \bar{\mathscr{A}}(i))$, where $\bar{\mathscr{E}}(i)$ and $\bar{\mathscr{A}}(i)$ are the corresponding edge set and adjacency matrix.

Based on the communication topology, let the extended tracking error for agent j at the ith iteration be

$$\xi_{i,j} = \sum_{k \in \mathscr{N}_j(i)} a_{j,k}(i)(\mathbf{y}_{i,k} - \mathbf{y}_{i,j}) + d_j(i)(\mathbf{y}_d - \mathbf{y}_{i,j}), \tag{3.3}$$

where $a_{j,k}(i)$ is the (j,k)th entry of the adjacency matrix $\mathscr{A}(i)$, $d_j(i) = 1$ if the jth agent can access the virtual leader's output at the ith iteration, and $d_j(i) = 0$ otherwise. The extended tracking error $\xi_{i,j}$ contains only local information. As such it can be used for distributed controller design.

The commonly applied D-type ILC rule is adopted in this chapter,

$$\mathbf{u}_{i+1,j} = \mathbf{u}_{i,j} + \Gamma_i \dot{\xi}_{i,j}, \tag{3.4}$$

where Γ_i is the learning gain to be designed.

Remark 3.1 *The derivative term of $\xi_{i,j}$ is utilized in the learning rule (3.4). Note that $\xi_{i,j}$ is only used in the $(i+1)$th iteration. Therefore, it is already available at the $(i+1)$th iteration. As measured signals are usually contaminated by noise, some sophisticated numerical method should be applied to obtain $\dot{\xi}_{i,j}$ without generating large amount of noise. For example, the filtered differentiation method in (Slotine and Li, 1991, pp.202) is a simple but effective candidate. In fact, such kind of non-causal implementation is one distinct feature of ILC.*

Before presenting the convergence properties of control law (3.4) under various communication assumptions, the identical initialization condition (*i.i.c.*) is imposed.

Assumption 3.1 *The initial condition of each agent is reset to the desired initial condition at every iteration, i.e.,* $\mathbf{x}_{i,j}(0) = \mathbf{x}_d(0)$.

The *i.i.c.* is the most commonly used assumption in ILC literature. *Remark* 2.4 discusses the relaxation or removal of *i.i.c.* at the cost of imperfect tracking performance. We will further investigate this issue in Chapter 4.

Initially we discuss the convergence properties under fixed communication topology, but eventually we will investigate the most general communication graph as below.

Definition 3.1 *Consider an iteration-varying graph* $\mathscr{G}(i)$. *The graph is said to be uniformly strongly connected along the iteration axis, if there exists a constant integer* K *such that the union graph* $\cup_{s=r}^{r+K}\mathscr{G}(s)$ *is strongly connected for all* r.

3.3 Main Results

In this section, convergence properties of control law (3.4) are investigated under three different communication assumptions. First, the convergence result is derived for fixed strongly connected graph in Section 3.3.1. The proof is detailed here and it will be referred by Subsections 3.3.2 and 3.3.3. Next the result is extended to iteration-varying strongly connected graph in Subsection 3.3.2. Finally, generalization to uniformly strongly connected graph is presented in Subsection 3.3.3.

3.3.1 Fixed Strongly Connected Graph

In this subsection, assume the communication topology is a fixed strongly connected graph, and at least one of the follower agents can access the leader's trajectory. The following two lemmas are introduced here as they will be utilized in the main proof.

Lemma 3.1 *For a given matrix M, if its spectral radius $\rho(M) < 1$, then there exist positive constants $c_1 > 0$, and $0 < \rho < 1$, such that $\|M^k\| \leq c_1\rho^k$.*

Proof: Since $\rho(M) < 1$, M is a stable matrix, i.e., M^k converges to zero exponentially as k goes to infinity. Therefore, it is straightforward to conclude Lemma 3.1. ■

In Lemma 3.1, c_1 is a positive constant. For convenience, choose $c_1 \geq 1$ in the following development, since when $c_1 < 1$, Lemma 3.1 still holds by setting $c_1 = 1$.

Lemma 3.2 *Consider a positive sequence $\{a_i\}$ satisfying that*

$$a_{i+1} = c_1\rho^i a_1 + \frac{c_2}{\lambda - L_f}(\rho^{i-1}a_1 + \rho^{i-2}a_2 + \cdots + a_i),$$

for $i \geq 1$, where L_f, c_1, and c_2 are positive constants, $0 < \rho < 1$. If $\lambda > L_f + \frac{c_2}{1-\rho}$, then $a_i \to 0$.

Proof: By hypothesis,

$$a_{i+1} = c_1\rho^i a_1 + \frac{c_2}{\lambda - L_f}(\rho^{i-1}a_1 + \rho^{i-2}a_2 + \cdots + a_i). \tag{3.5}$$

Then, for $i \geq 2$, we have

$$a_i = c_1\rho^{i-1}a_1 + \frac{c_2}{\lambda - L_f}(\rho^{i-2}a_1 + \rho^{i-3}a_2 + \cdots + a_{i-1}), \tag{3.6}$$

From (3.5) and (3.6) we can obtain,

$$\begin{aligned} a_{i+1} - \rho a_i &= \frac{c_2}{\lambda - L_f}a_i, \\ a_{i+1} &= \left(\rho + \frac{c_2}{\lambda - L_f}\right)a_i. \end{aligned}$$

As $\lambda > L_f + \frac{c_2}{1-\rho}$, we have $\rho + \frac{c_2}{\lambda - L_f} < 1$. Hence, $a_i \to 0$. ■

Let the learning gain be identical for all iterations since the communication graph is fixed,

$$\Gamma = \frac{1}{q}[(CB)^T CB]^{-1}(CB)^T, \tag{3.7}$$

where

$$q > \max_{j=1\dots N} \sum_{k=1}^{N} a_{j,k} + d_j.$$

To be more accurate, $a_{j,k}$ and d_j in the inequality above are supposed to be $a_{j,k}(i)$ are $d_j(i)$. As the graph is iteration-invariant, the iteration index i is omitted.

Now define the actual tracking error $\mathbf{e}_{i,j} = \mathbf{y}_d - \mathbf{y}_{i,j}$, together with (3.3), the control law (3.4) can be rewritten in a compact form,

$$\mathbf{u}_{i+1} = \mathbf{u}_i + (H \otimes \Gamma)\dot{\mathbf{e}}_i, \tag{3.8}$$

where $\mathbf{u}_i$ and $\mathbf{e}_i$ are the column stack vectors of $\mathbf{u}_{i,j}$ and $\mathbf{e}_{i,j}$, $H = L + D$, L is the Laplacian matrix of $\mathscr{G}$, $D = \mathrm{diag}(d_1, d_2, \dots, d_N)$, and $\otimes$ denotes the Kronecker product.

Theorem 3.1 *Consider the multi-agent systems (3.1) under Assumption 3.1, the learning gain (3.7), and control law (3.8). If the communication topology is a fixed strongly connected graph, and at least one of the followers in the network has access to the virtual leader's trajectory, then the tracking error $\mathbf{e}_{i,j}$ converges to zero along the iteration axis, i.e.,* $\lim_{i\to\infty} \mathbf{e}_{i,j} = 0$.

Proof: Define $\delta\mathbf{u}_{i,j} = \mathbf{u}_d - \mathbf{u}_{i,j}$, $\delta\mathbf{x}_{i,j} = \mathbf{x}_d - \mathbf{x}_{i,j}$, and $\delta\mathbf{f}(\mathbf{x}_{i,j}) = \mathbf{f}(\mathbf{x}_d) - \mathbf{f}(\mathbf{x}_{i,j})$. Subtracting (3.1) from (3.2) yields

$$\delta\dot{\mathbf{x}}_{i,j} = \delta\mathbf{f}(\mathbf{x}_{i,j}) + B\delta\mathbf{u}_{i,j}. \tag{3.9}$$

Let $\delta\mathbf{u}_i$, $\delta\mathbf{x}_i$, and $\delta\mathbf{f}(\mathbf{x}_i)$ be the column stack vectors of $\delta\mathbf{u}_{i,j}$, $\delta\mathbf{x}_{i,j}$, and $\delta\mathbf{f}(\mathbf{x}_{i,j})$, respectively. Then, (3.9) can be written as

$$\delta\dot{\mathbf{x}}_i = \delta\mathbf{f}(\mathbf{x}_i) + (I_N \otimes B)\delta\mathbf{u}_i, \tag{3.10}$$

where I is the identity matrix, and the subscript denotes its dimension.

Let us investigate the variation of $\delta\mathbf{u}_i$ between two consecutive iterations. From the learning rule (3.8), we have

$$\begin{aligned}\delta\mathbf{u}_{i+1} &= \delta\mathbf{u}_i - (H\otimes\Gamma)\dot{\mathbf{e}}_i,\\ &= \delta\mathbf{u}_i - (H\otimes\Gamma)(I_N\otimes C)\delta\dot{\mathbf{x}}_i. \end{aligned}\tag{3.11}$$

Substitute (3.10) to (3.11), and notice the learning gain (3.7), then we can obtain

$$\delta\mathbf{u}_{i+1} = \left(\left(I_N - \frac{1}{q}H\right)\otimes I_p\right)\delta\mathbf{u}_i - (H\otimes\Gamma C)\delta\mathbf{f}(\mathbf{x}_i). \tag{3.12}$$

For simplicity, denote $M = \left(I_N - \frac{1}{q}H\right)\otimes I_p$, and $F = H\otimes\Gamma C$. From (3.12), we can obtain the relation between $\delta\mathbf{u}_{i+1}$ and $\delta\mathbf{u}_1$ below

$$\delta\mathbf{u}_{i+1} = M^i\delta\mathbf{u}_1 - M^{i-1}F\delta\mathbf{f}(\mathbf{x}_1) - M^{i-2}F\delta\mathbf{f}(\mathbf{x}_2) - \cdots - F\delta\mathbf{f}(\mathbf{x}_i). \tag{3.13}$$

Since the communication graph is strongly connected, $\left(I_N - \frac{1}{q}H\right)$ must be an irreducible matrix. Notice that q is larger than the greatest diagonal entry of H, and D contains at least one positive entry, hence, at least one row sum of $\left(I_N - \frac{1}{q}H\right)$ is strictly less than one. Therefore, $\left(I_N - \frac{1}{q}H\right)$ is a substochastic matrix. By using the *Corollary 6.2.28* in (Horn and Johnson, 1985, pp.363), we can conclude that the spectral radius $\rho\left(I_N - \frac{1}{q}H\right) < 1$. Note that M and $\left(I_N - \frac{1}{q}H\right)$ have the same spectrum, except that each eigenvalue of M has multiplicity of p. Therefore, $\rho(M) < 1$. Take infinity norm of (3.13), and apply global Lipschitz condition on $\mathbf{f}(\cdot)$ and Lemma 3.1, we have

$$|\delta\mathbf{u}_{i+1}| \leq c_1\rho^i|\delta\mathbf{u}_1| + c_1L_f\rho^{i-1}|F||\delta\mathbf{x}_1| + c_1L_f\rho^{i-2}|F||\delta\mathbf{x}_2| + \cdots + L_f|F||\delta\mathbf{x}_i|, \tag{3.14}$$

where $0 < \rho < 1$ and satisfies the inequality in Lemma 3.1.

Integrating (3.10) together with Assumption 3.1 yields

$$\delta\mathbf{x}_i \leq \int_0^t \left(\delta\mathbf{f}(\mathbf{x}_i) + (I_N\otimes B)\delta\mathbf{u}_i\right)d\tau. \tag{3.15}$$

Taking norm operations on both sides of (3.15), and applying the Gronwall-Bellman's Lemma, we have

$$|\delta \mathbf{x}_i| \leq \int_0^t e^{L_f(t-\tau)} |I_N \otimes B| |\delta \mathbf{u}_i| \, d\tau. \tag{3.16}$$

Taking λ-norm of (3.16) yields

$$\|\delta \mathbf{x}_i\|_\lambda \leq \frac{1}{\lambda - L_f} \|I_N \otimes B\| \|\delta \mathbf{u}_i\|_\lambda. \tag{3.17}$$

Taking λ-norm of (3.14), together with (3.17), yields

$$\begin{aligned}
&\|\delta \mathbf{u}_{i+1}\|_\lambda \\
&\leq c_1 \rho^i \|\delta \mathbf{u}_1\|_\lambda + c_1 \rho^{i-1} \frac{L_f}{\lambda - L_f} \|I_N \otimes B\| \|F\| \|\delta \mathbf{u}_1\|_\lambda \\
&+ c_1 \rho^{i-2} \frac{L_f}{\lambda - L_f} \|I_N \otimes B\| \|F\| \|\delta \mathbf{u}_2\|_\lambda + \cdots + \frac{L_f}{\lambda - L_f} \|I_N \otimes B\| \|F\| \|\delta \mathbf{u}_i\|_\lambda.
\end{aligned} \tag{3.18}$$

We can purposely choose $c_1 \geq 1$, and denote $c_2 = c_1 L_f \|I_N \otimes B\| \|F\|$, thus, we have

$$\begin{aligned}
&\|\delta \mathbf{u}_{i+1}\|_\lambda \\
&\leq c_1 \rho^i \|\delta \mathbf{u}_1\|_\lambda + \frac{c_2}{\lambda - L_f} \rho^{i-1} \|\delta \mathbf{u}_1\|_\lambda \\
&+ \frac{c_2}{\lambda - L_f} \rho^{i-2} \|\delta \mathbf{u}_2\|_\lambda + \cdots + \frac{c_2}{\lambda - L_f} \|\delta \mathbf{u}_i\|_\lambda.
\end{aligned} \tag{3.19}$$

Choose a $\lambda > L_f + \frac{c_2}{1-\rho}$. Applying Lemma 3.2 and *comparison Lemma*, it can be concluded that $\delta \mathbf{u}_{i,j} \to 0$ along the iteration axis, hence, $\lim_{i \to \infty} \mathbf{e}_{i,j} = 0$. ■

Remark 3.2 *From the proof to Lemma 3.2, the sequence $\{a_i\}$ converges to zero exponentially. As Theorem 3.1 is proved by using Lemma 3.2, the λ-norm of $\delta \mathbf{u}_i$, $\|\delta \mathbf{u}_i\|_\lambda$, should also converge to zero exponentially. From the definition of λ-norm, it can be shown that $\|\delta \mathbf{u}_i\| \leq e^{\lambda T} \|\delta \mathbf{u}_i\|_\lambda$. Note that $e^{\lambda T}$ is a constant. Therefore, $\|\delta \mathbf{u}_i\|$ and the tracking error $\|\mathbf{e}_i\|$ both converge to zero exponentially as well.*

3.3.2 Iteration-varying Strongly Connected Graph

In this subsection, we assume the communication topology is iteration-varying, but the graph is fixed and strongly connected in each iteration. Furthermore, at least one of the followers has access to the leader's trajectory in every iteration.

We adopt the following iteration-varying learning gain

$$\Gamma(i) = \frac{1}{q(i)} \left((CB)^T CB\right)^{-1} (CB)^T, \tag{3.20}$$

where $\Gamma(i)$ depends on the iteration index i, and

$$q(i) > \max_{j=1\dots N} \sum_{k=1}^{N} a_{j,k}(i) + d_j(i).$$

ILC rule (3.8) becomes

$$\mathbf{u}_{i+1} = \mathbf{u}_i + (H(i) \otimes \Gamma(i))\dot{\mathbf{e}}_i. \tag{3.21}$$

Since the communication topology is iteration-varying, the system matrix M in (3.12) is iteration-varying as well, and Lemma 3.1 is no longer applicable to show the convergence result. Therefore, we develop two preliminary results first before presenting the main convergence property.

Lemma 3.3 *Let $\mathscr{M} \subset \mathbb{R}^{N\times N}$ denote the set of all irreducible substochastic matrices with positive diagonal entries, then we have*

$$|M(N)M(N-1)\cdots M(1)| < 1,$$

where $M(k)$, $k = 1,2,\dots,N$, are N matrices arbitrarily selected from $\mathscr{M}$.

Proof: We show the result by induction. Let $\mathbf{1}_{(\cdot)}$ be a vector with all elements being 1. The subscript $(\cdot)$ denotes its dimension, and the subscript is omitted when the dimension is clear in the context.

Since the $M(k)$ is a substochastic matrix, therefore, at least one element in the vector $M(1)\mathbf{1}$ must be strictly less than 1.

Next, assume that multiplication of k matrices from $\mathscr{M}$ has k row sums less than 1. Without loss of generality, assume the first k row sums less than 1, that is

$$M(k)M(k-1)\cdots M(1)\mathbf{1} = \begin{bmatrix} \alpha \\ \mathbf{1}_{N-k} \end{bmatrix},$$

where $\alpha \in \mathbb{R}^k$ with all elements less than 1.

Then, investigate the multiplication of $k+1$ matrices from $\mathscr{M}$. Set

$$\begin{bmatrix} M_{11}(k+1) & M_{12}(k+1) \\ M_{21}(k+1) & M_{22}(k+1) \end{bmatrix} \begin{bmatrix} \alpha \\ \mathbf{1}_{N-k} \end{bmatrix} = \begin{bmatrix} \beta \\ \gamma \end{bmatrix}.$$

Since the diagonal entries of $M_{11}(k+1)$ are positive, all the elements in β have to be less than 1. If $\gamma = \mathbf{1}_{N-k}$, it implies $M_{21}(k+1)$ is a zero block matrix, and it contradicts to the fact $M(k+1)$ is irreducible. So multiplication of $k+1$ matrices from $\mathscr{M}$ must have at least $k+1$ row sums less than 1. This completes the proof. ■

Lemma 3.4 *Consider a positive sequence $\{a_i\}$ satisfying that*

$$a_{i+1} = \rho^{\lfloor \frac{i}{N} \rfloor} a_1 + \frac{c_2}{\lambda - L_f}\left(\rho^{\lfloor \frac{i-1}{N} \rfloor} a_1 + \rho^{\lfloor \frac{i-2}{N} \rfloor} a_2 + \cdots + \rho^{\lfloor \frac{i-i}{N} \rfloor} a_i\right),$$

for $i \geq 1$, where $\lfloor \cdot \rfloor$ stands for the floor function, L_f and c_2 are positive constants, $0 < \rho < 1$. If $\rho + \left(1 + \frac{c_2}{\lambda - L_f}\right)^N - 1 < 1$, then $a_i \to 0$.

Proof: First we investigate how a_{i+1} evolves for $i = 1, 2, \ldots, N-1$. By hypothesis, we have

$$a_{i+1} = a_1 + \frac{c_2}{\lambda - L_f}(a_1 + a_2 + \cdots + a_i),$$

and

$$a_i = a_1 + \frac{c_2}{\lambda - L_f}(a_1 + a_2 + \cdots + a_{i-1}).$$

Therefore, we have

$$a_{i+1} = \left(1 + \frac{c_2}{\lambda - L_f}\right) a_i,\ i = 1, 2, \ldots, N-1.$$

Subsequently, we can find a general formula for a_{i+1},

$$a_{i+1} = \left(1 + \frac{c_2}{\lambda - L_f}\right)^i a_1,\ i = 1, 2, \ldots, N-1. \tag{3.22}$$

Next, let us study the relation between a_{N+1} and a_1.

$$a_{N+1} = \rho a_1 + \frac{c_2}{\lambda - L_f}(a_1 + a_2 + \cdots + a_N). \tag{3.23}$$

Noticing (3.22), equation (3.23) can be simplified below

$$\begin{aligned} a_{N+1} &= \rho a_1 + \frac{c_2}{\lambda - L_f}\left(1 + \left(1 + \frac{c_2}{\lambda - L_f}\right) + \left(1 + \frac{c_2}{\lambda - L_f}\right)^2 \right. \\ &\quad \left. + \cdots + \left(1 + \frac{c_2}{\lambda - L_f}\right)^{N-1}\right) a_1 \\ &= \rho a_1 + \frac{c_2}{\lambda - L_f} \frac{\left(1 + \frac{c_2}{\lambda - L_f}\right)^N - 1}{\left(1 + \frac{c_2}{\lambda - L_f}\right) - 1} a_1 \\ &= (\rho + \left(1 + \frac{c_2}{\lambda - L_f}\right)^N - 1) a_1. \end{aligned}$$

By using the similar procedure above, eventually, we can obtain that

$$a_{kN+1} = \left(\rho + \left(1 + \frac{c_2}{\lambda - L_f}\right)^N - 1\right)^k a_1.$$

Hence, if $\rho + (1 + \frac{c_2}{\lambda - L_f})^N - 1 < 1$, $a_i \to 0$. ■

The next theorem shows the convergence result for iteration-varying communication.

Theorem 3.2 *Consider the multi-agent systems (3.1) under Assumption 3.1, the learning gain (3.20), and control law (3.21). If the communication graph is iteration-varying, but in each iteration, the graph is fixed and strongly connected, and at least one of the followers in the network has access to the virtual leader's trajectory, then the tracking error* $\mathbf{e}_{i,j}$ *converges to zero along the iteration axis, i.e.,* $\lim_{i \to \infty} \mathbf{e}_{i,j} = 0$.

Proof: Denote $M(i) = \left(I_N - \frac{1}{q(i)}H(i)\right) \otimes I_p$, and $F(i) = H(i) \otimes \Gamma(i)C$. For notational simplicity, define $M(i,k) = M(i)M(i-1)\cdots M(k)$ for $i \geq k$.

Follow the analysis framework in the proof of Theorem 3.1, we can get

$$\begin{aligned} \delta\mathbf{u}_{i+1} &= M(i,1)\delta\mathbf{u}_1 - M(i-1,1)F(1)\delta\mathbf{f}(\mathbf{x}_1) \\ &\quad -M(i-2,2)F(2)\delta\mathbf{f}(\mathbf{x}_2) - \cdots - F(i)\delta\mathbf{f}(\mathbf{x}_i). \end{aligned} \tag{3.24}$$

From the definition of $\Gamma(i)$ in (3.20), we can show that $M(i)$ is a substochastic matrix with positive diagonal entries. Take infinity norm on both sides of equation (3.24), we can obtain

$$\begin{aligned} |\delta\mathbf{u}_{i+1}| &\leq |M(i,1)||\delta\mathbf{u}_1| + L_f|M(i-1,1)||F(1)||\delta\mathbf{x}_1| \\ &\quad +L_f|M(i-2,2)||F(2)||\delta\mathbf{x}_2| + \cdots + L_f|F(i)||\delta\mathbf{x}_i|. \end{aligned} \tag{3.25}$$

Group every N matrices product together in (3.25), and apply Lemma 3.3, we have

$$\begin{aligned} |\delta\mathbf{u}_{i+1}| &\leq \rho^{\lfloor \frac{i}{N} \rfloor}|\delta\mathbf{u}_1| + L_f\rho^{\lfloor \frac{i-1}{N} \rfloor}|F(1)||\delta\mathbf{x}_1| \\ &\quad +L_f\rho^{\lfloor \frac{i-2}{N} \rfloor}|F(2)||\delta\mathbf{x}_2| + \cdots + L_f|F(i)||\delta\mathbf{x}_i|. \end{aligned} \tag{3.26}$$

Taking λ-norm of (3.26), and substituting in equation (3.17) yield,

$$\begin{aligned} \|\delta\mathbf{u}_{i+1}\|_\lambda &\leq \rho^{\lfloor \frac{i}{N} \rfloor}\|\delta\mathbf{u}_1\|_\lambda + \frac{L_f}{\lambda - L_f}\|I_N \otimes B\|\rho^{\lfloor \frac{i-1}{N} \rfloor}\|F(1)\|\|\delta\mathbf{u}_1\|_\lambda \\ &\quad + \frac{L_f}{\lambda - L_f}\|I_N \otimes B\|\rho^{\lfloor \frac{i-2}{N} \rfloor}\|F(2)\|\|\delta\mathbf{u}_2\|_\lambda \\ &\quad + \cdots + \frac{L_f}{\lambda - L_f}\|I_N \otimes B\|\|F(i)\|\|\delta\mathbf{u}_i\|_\lambda. \end{aligned} \tag{3.27}$$

Let $c_2 = \max_{k=1}^{i} L_f\|F(k)\|\|I_N \otimes B\|$, (3.27) becomes

$$\begin{aligned} \|\delta\mathbf{u}_{i+1}\|_\lambda \leq\ & \rho^{\lfloor \frac{i}{N} \rfloor}\|\delta\mathbf{u}_1\|_\lambda + \frac{c_2}{\lambda - L_f}\left(\rho^{\lfloor \frac{i-1}{N} \rfloor}\|\delta\mathbf{u}_1\|_\lambda + \rho^{\lfloor \frac{i-2}{N} \rfloor}\|\delta\mathbf{u}_2\|_\lambda + \cdots \right. \\ & \left. +\rho^{\lfloor \frac{i-i}{N} \rfloor}\|\delta\mathbf{u}_i\|_\lambda\right), \end{aligned}$$

Based on the result in Lemma 3.4, we can conclude that $\delta\mathbf{u}_{i,j} \to 0$ along the iteration axis, hence, $\lim_{i\to\infty} \mathbf{e}_{i,j} = 0$ as well. ■

3.3.3 Uniformly Strongly Connected Graph

In this subsection, we further generalize the results in the previous two subsections. Let the communication be the uniformly strongly connected graph as defined in *Definition* 3.1. Such a condition is much more flexible than the communication assumptions in Theorem 3.1 and Theorem 3.2. The following Lemma is required to generalize our results obtained previously.

Lemma 3.5 *Let $i \geq 2$ be an integer and let $Q_1, Q_2, \ldots, Q_i \in \mathbb{R}^{N\times N}$ be nonnegative matrices. Suppose the diagonal entries of Q_k $(k = 1, 2, \ldots, i)$ are positive, then there exists a $\gamma > 0$ such that*

$$Q_1 Q_2 \cdots Q_i \succeq \gamma (Q_1 + Q_2 + \cdots + Q_i),$$

where $\succeq$ is defined entrywise.

The proof of Lemma 3.5 can be found in Jadbabaie et al. (2003).

The theorem below presents the most general convergence result in this chapter.

Theorem 3.3 *Consider the multi-agent systems (3.1) under Assumption 3.1, the learning gain (3.20), and control law (3.21). If the communication graph is uniformly strongly connected along the iteration axis with upper bound K, and in every K consecutive iterations, if there is at least one of the followers in the network having access to the virtual leader's trajectory, then the tracking error $\mathbf{e}_{i,j}$ converges to zero along the iteration axis, i.e.,* $\lim_{i\to\infty} \mathbf{e}_{i,j} = 0$.

Proof: Let $\tilde{M}(i) = I_N - \frac{1}{q(i)} H(i)$. $\tilde{M}(i)$ is a nonnegative matrix with positive diagonal entries. Since the communication graph is uniformly strongly connected along the iteration axis with upper bound K, and applying Lemma 3.5, we can get that $\tilde{M}(i+K, i)$ is an irreducible nonnegative matrix for any i. Furthermore, if there is at least one follower

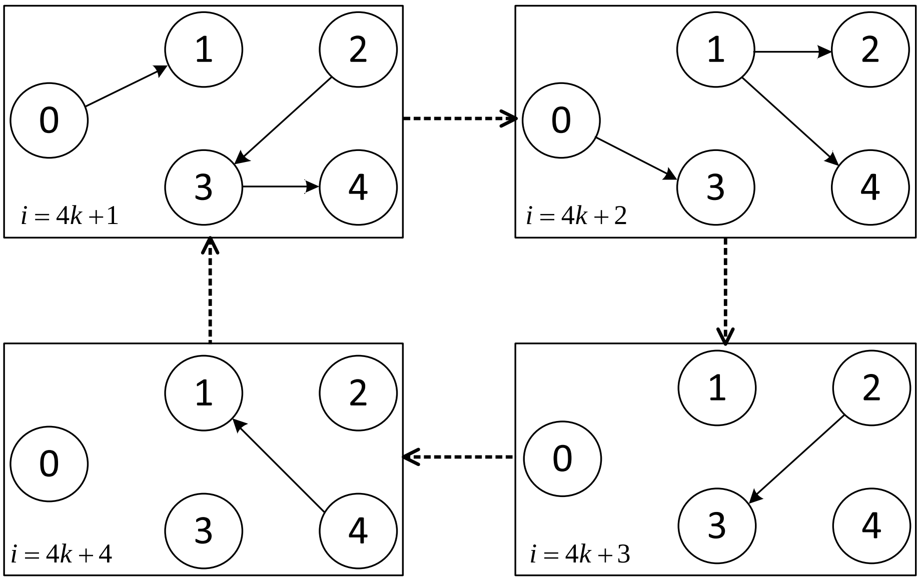

Figure 3.1: Communication topology among agents in the network.

in network having access to the leader's trajectory within the iteration interval $[i, i+K]$, then $\tilde{M}(i+K, i)$ is a substochastic matrix.

Now follow the same idea in the proof of Theorem 3.2, eventually, we can obtain that

$$\begin{aligned} \|\delta\mathbf{u}_{i+1}\|_\lambda \;\leq\; & \rho^{\lfloor \frac{i}{NK} \rfloor}\|\delta\mathbf{u}_1\|_\lambda + \frac{c_2}{\lambda - L_f}\Big(\rho^{\lfloor \frac{i-1}{NK} \rfloor}\|\delta\mathbf{u}_1\|_\lambda + \rho^{\lfloor \frac{i-2}{NK} \rfloor}\|\delta\mathbf{u}_2\|_\lambda + \cdots \\ & + \rho^{\lfloor \frac{i-i}{NK} \rfloor}\|\delta\mathbf{u}_i\|_\lambda\Big). \end{aligned}$$

Therefore, if $\rho + (1+\frac{c_2}{\lambda - L_f})^{NK} - 1 < 1$, according to Lemma 3.4, we can conclude that the tracking error $\mathbf{e}_{i,j}$ converges to zero as iteration goes to infinity. ■

Remark 3.3 *Intuitively, the convergence rate will become slow when the communication is weak, for instance the uniformly strongly connected graph assumption. This point can be verified in the proof of Theorem 3.3. If the upper bound K is large, the worst convergence rate of $\|\delta\mathbf{u}_i\|_\lambda$ becomes slow.*

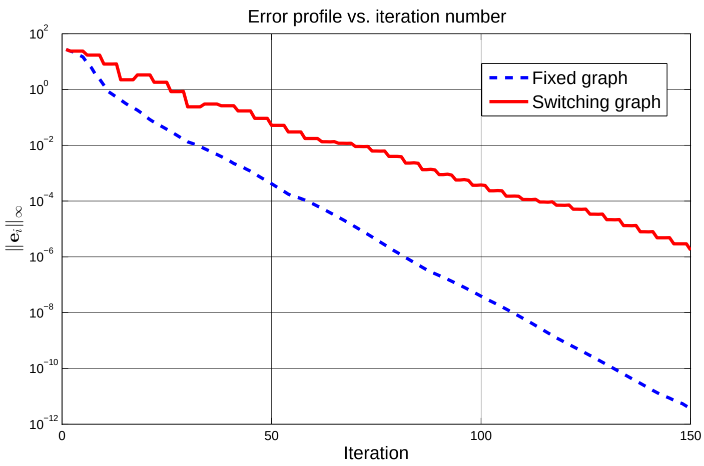

Figure 3.2: Maximum norm of error vs. iteration number.

3.4 Illustrative Example

The results in *Theorems* 3.1 and 3.2 are the special cases of Theorem 3.3. To test the theorems, it is sufficient to verify the result in Theorem 3.3. Hence, a tracking example under the uniformly strongly connected topology is provided here. Consider four dynamic agents, and their system parameters are given by

$$\mathbf{f}(\mathbf{x}) = \begin{bmatrix} x_2 \\ \cos(x_1) - x_2 \end{bmatrix}, \quad B = \begin{bmatrix} 0 \\ 1 \end{bmatrix}, \quad C = \begin{bmatrix} 1 & 0 \\ 2 & 1 \end{bmatrix}.$$

The leader's input $u_d = t + 4\sin(2t)$, $t \in [0,5]$. u_d is unknown to any of the followers.

The complete communication topology is depicted in Figure. 3.1, in which there are four graphs. In each iteration, only one of the communication graph is activated, and it is chosen by the selection function $i = 4k + j$, $j = 1,2,3,4$, and k is a nonnegative integer, for example, if $i = 1$, the graph on the left top corner is selected. In each of the graph, the topology is not even connected. However, the union of all the four graphs is indeed strongly connected. Therefore, the communication topology is uniformly strongly connected along the iteration axis with upper bound $K = 4$. Thus, the communication requirement in Theorem 3.3 is satisfied. To design the learning gain, we need to investigate the matrices $H(i) = L(i) + D(i)$, and the first four of them are listed below.

$$H(1) = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & -1 & 1 & 0 \\ 0 & 0 & -1 & 1 \end{bmatrix}, H(2) = \begin{bmatrix} 0 & 0 & 0 & 0 \\ -1 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ -1 & 0 & 0 & 1 \end{bmatrix}$$

$$H(3) = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & -1 & 1 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}, H(4) = \begin{bmatrix} 1 & 0 & 0 & -1 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}$$

The maximal diagonal entry of $H(i)$ is 1, hence, we can choose the learning gain $q_s = 2$ for switching topology.

Denote the union graph of the four subgraphs in Figure. 3.1 by $\mathscr{G}_u$. The corresponding graph Laplacian L_u and matrix D_u are

$$L_u = \begin{bmatrix} 1 & 0 & 0 & -1 \\ -1 & 1 & 0 & 0 \\ 0 & -1 & 1 & 0 \\ -1 & 0 & -1 & 2 \end{bmatrix}$$

and $D_u = \text{diag}(1, 0, 1, 0)$ respectively. As a comparative study, we also investigate the convergence performance of the four agents under the fixed graph $\mathscr{G}_u$. In this case, let the learning gain $q_f = 3$. Notice that q_s and q_f are slightly different. This is because we choose the value such that the control performances are the best for two cases respectively. Figure. 3.2 describes how the maximum error $\|\mathbf{e}_i\|_\infty = \max_{t\in[0,T]} |\mathbf{e}_i(t)|$ evolves along the iteration axis for both switching and fixed graphs. The y-axis is plotted in log-scale. That means the convergence rates in both cases are exponentially fast. It can be seen from Figure. 3.2 that the dashed line is below the solid line. The observation indicates that the intermittent communication slows down the convergence rate. This also shows the simulation results match perfectly with our theoretical predictions.

3.5 Conclusion

In this chapter, a typical D-type ILC rule for consensus tracking is investigated under iteration-varying graph. It is shown that if the iteration-varying graph is uniformly strongly connected along the iteration axis, and at least one of the followers has access to the leader's trajectory, the proposed ILC rule can perfectly synchronize the output trajectories of all follower agents. The developed results complement the existing literature by generalizing the fixed communication to iteration-varying graph. Simulation study demonstrates that the learning controller is very robust to communication variations.

Chapter 4

Iterative Learning Control for Multi-agent Coordination with Initial State Error

4.1 Background

In the discussions of Chapters 2 and 3, the perfect identical initial condition (*i.i.c.*) is assumed. Similarly, most existing ILC for multi-agent systems coordination assume this restrictive assumption, for example Ahn and Chen (2009); Xu et al. (2011); Yang et al. (2012); Yang and Xu (2012); Liu and Jia (2012); Meng et al. (2012); Li and Li (2013). Notice that the *i.i.c.* is one of the fundamental problems pertaining to ILC literature and its applicability. It is required that the initial state is the same as the desired initial state for perfect tracking to be achievable. However, the controllers in the multi-agent systems are distributed in nature, and all the agents are independent entities. The communication among agents may not be complete so that many agents are not aware of the desired initial state. It is difficult to ensure perfect *i.i.c.* for all agents in general. Thus, the initial condition problem in multi-agent systems requires further investigation.

There are many excellent ideas in ILC literature to deal with the imperfect initial conditions. Many researchers have analyzed and developed algorithms that do not require the *i.i.c.* at the cost of imperfect tracking. The initial state learning method is developed in Chen et al. (1999), which is applicable when the initial state is measurable and manipulatable. Some extra system knowledge is required by the learning rule. When the learning gain is chosen correctly, the initial state converges to the desired one, and the tracking error due to initial state error is bounded. The initial rectifying action control is introduced by Sun and Wang (2002), which modifies the desired trajectory such that the initial output of the new reference coincides with the actual initial output. Because of the nature of this control method, the actual output converges to a modified desired trajectory. These two approaches are adopted for multi-agent control with initial state error, for example initial state learning is applied in Yang et al. (2012), and initial rectifying action is utilized in Meng et al. (2012). Variable initial state for discrete-time system is discussed in Fang and Chow (2003) via 2-D analysis. Park et al. (1999) study the robustness of PID-type updating rule. In Park (2005) an average operator based PD-type rule is developed to improve the control performance against variable initial state error. Besides, Xu and Yan (2005) fully investigate five different initial conditions and their convergence properties.

In this chapter, we investigate the initial state error and its impact on the control performance in the multi-agent consensus tracking problem. At the beginning of a task execution, the initial state of each agent is reset to a fixed position which is different from the desired one. Under sparse communication assumption, it is shown that the D-type (Xu and Tan, 2003) ILC rule is still applicable and convergence can be guaranteed when the communication graph contains a spanning tree with the leader being its root. However, in terms of performance the final output of each agent has large deviations from the desired reference. To improve the performance, PD-type (Park et al., 1999) rule is motivated, and it turns out that the new updating rule gives the designer more freedom to tune the final performance. The analysis method itself in this work is of independent interest, and it contributes to the ILC literature on the initial state problem.

This chapter is organized as follows. The consensus tracking problem with initial state error is formulated in Section 4.2. Next, the D-type, PD-type updating rules, and convergence results are developed in Section 4.3. To demonstrate the effectiveness of the results, two numerical examples are presented in Section 4.4. Lastly, Section 4.5 draws the conclusion.

4.2 Problem Description

Consider a group of N homogeneous dynamic agents, and the jth agent is governed by the following linear time-invariant model,

$$\left\{ \begin{array}{l} \dot{\mathbf{x}}_{i,j}(t) = A\mathbf{x}_{i,j}(t) + B\mathbf{u}_{i,j}(t) \\ \mathbf{y}_{i,j}(t) = C\mathbf{x}_{i,j}(t) \end{array} \right. \quad \forall j \in \mathscr{V}, \tag{4.1}$$

where i denotes the iteration number, $\mathbf{x}_{i,j} \in \mathbb{R}^n$, $\mathbf{y}_{i,j} \in \mathbb{R}^p$, and $\mathbf{u}_{i,j} \in \mathbb{R}^m$ are the state vector, output vector, and control input respectively, and A, B, C are constant matrices of compatible dimensions. For simplicity, the time argument, t, is dropped when no confusion arises.

The leader's trajectory, or the desired consensus trajectory $\mathbf{y}_d(t)$ is defined on a finite-time interval $[0,T]$, and it is generated by the following dynamics,

$$\begin{cases} \dot{\mathbf{x}}_d = A\mathbf{x}_d + B\mathbf{u}_d, \\ \mathbf{y}_d = C\mathbf{x}_d, \end{cases} \tag{4.2}$$

where $\mathbf{u}_d$ is the continuous and unique desired control input.

Due to communication or sensor limitations, the leader's trajectory is only accessible to a small portion of the followers. Let the communication among followers be described by the graph $\mathscr{G}(\mathscr{V},\mathscr{E})$. If the leader is labeled by vertex 0, then the complete information flow among all the agents can be characterized by a new graph $\overline{\mathscr{G}} = (0 \cup \mathscr{V}, \overline{\mathscr{E}})$, where $\overline{\mathscr{E}}$ is the new edge set. The major task is to design a set of distributed ILC rules such that each individual agent in the network is able to track the leader's trajectory under the sparse communication graph $\overline{\mathscr{G}}$.

To simplify the controller design and convergence analysis, the following two assumptions are imposed.

Assumption 4.1 *CB is of full column rank.*

Remark 4.1 *The full column rank assumption is a necessary condition to find out a suitable learning gain such that ILC D-type learning rule satisfies the contraction-mapping criterion. Assumption 4.1 implies that the relative degree of system (4.1) is well defined and it is exactly 1. When CB is not of full rank, high-order derivative of the tracking error can be utilized in the controller design.*

Assumption 4.2 *The initial state of an agent is reset to the same position at every iteration, which is not equal to the desired state, i.e.,* $\mathbf{x}_{i,j}(0) = \mathbf{x}_{1,j}(0) \neq \mathbf{x}_d(0)$ *for all* $i \geq 1$.

Remark 4.2 *Assumption 4.2 is referred as the resetting condition in ILC literature. Since the initial state is different from the desired state, that is* $\mathbf{y}_{i,j}(0) \neq \mathbf{y}_d(0)$*, it is im-*

possible to achieve the perfect tracking. The ILC rule should force the output trajectory of each agent to be as close as possible to the leader's trajectory.

4.3 Main Results

The main results contain two sections. In Section 4.3.1, the distributed D-type updating rule and its convergence properties are fully analyzed. To improve the tracking performance of learning rules, PD-type updating rule is proposed.

4.3.1 Distributed D-type Updating Rule

Let $\xi_{i,j}$ be the distributed measurement by agent j at the ith iteration over the graph $\overline{\mathscr{G}}$, and it is defined as

$$\xi_{i,j} = \sum_{k\in\mathscr{N}_j} a_{j,k}(\mathbf{y}_{i,k} - \mathbf{y}_{i,j}) + d_j(\mathbf{y}_d - \mathbf{y}_{i,j}), \tag{4.3}$$

where $d_j = 1$ if $(0, j) \in \overline{\mathscr{E}}$, otherwise $d_j = 0$.

Note that the actual tracking error $\mathbf{e}_{i,j} = \mathbf{y}_d - \mathbf{y}_{i,j}$ cannot be utilized in the controller design as only a small number of followers have access to the leader's trajectory. Therefore, $\mathbf{e}_{i,j}$ is not available to many of the followers. It is natural to incorporate the distributed measurement $\xi_{i,j}$ in the ILC design. Hence, the following D-type ILC updating rule is adopted in this work,

$$\mathbf{u}_{i+1,j} = \mathbf{u}_{i,j} + Q\dot{\xi}_{i,j},\ \mathbf{u}_{0,j} = 0,\ \forall j \in \mathscr{V}, \tag{4.4}$$

where Q is the learning gain to be designed. For simplicity, the initial control input $\mathbf{u}_{0,j}$ is set to zero. However, in practical implementation, the initial control input can be generated by certain feedback mechanism such that the system is stable. This may improve the transient performance of the learning controller. Note that $\xi_{i,j}$ is already available at the $(i+1)$th iteration. Therefore, the derivative of $\xi_{i,j}$ can be obtained by any sophisticated numerical differentiation that does not generate large amount of noise.

Let $\hat{\mathbf{x}}_{v,j}$, $\hat{\mathbf{u}}_{v,j}$, and $\hat{\mathbf{y}}_{v,j} = C\hat{\mathbf{x}}_{v,j}$ satisfy the following virtual dynamics,

$$\hat{\mathbf{x}}_{v,j}(t) = \mathbf{x}_{i,j}(0) + \int_0^t [A\hat{\mathbf{x}}_{v,j}(\tau) + B\hat{\mathbf{u}}_{v,j}(\tau)]\, d\tau, \tag{4.5}$$

and

$$Q(\dot{\mathbf{y}}_d - \dot{\hat{\mathbf{y}}}_{v,j}) = 0. \tag{4.6}$$

Before investigating the existence and uniqueness of $\hat{\mathbf{u}}_{v,j}$, $\hat{\mathbf{x}}_{v,j}$, and $\hat{\mathbf{y}}_{v,j}$ satisfying equations (4.5) and (4.6), we have the following convergence results.

Theorem 4.1 *Consider the multi-agent systems (4.1), under Assumptions 4.1, 4.2, the communication graph $\overline{\mathcal{G}}$, and distributed D-type updating rule (4.4). If the learning gain Q is chosen such that*

$$|I_{mN} - H \otimes QCB| \leq \mu < 1,$$

where $I_{(\cdot)}$ is the identity matrix with the subscript denoting its dimension, μ is a constant, $H = L + D$, L is the Laplacian matrix of $\mathcal{G}$, $D = diag(d_1, d_2, \ldots, d_N)$, and $\otimes$ represents the Kronecker product, then the control input $\mathbf{u}_{i,j}$ and output $\mathbf{y}_{i,j}$ converge to $\hat{\mathbf{u}}_{v,j}$ and $\hat{\mathbf{y}}_{v,j}$ respectively as iteration goes to infinity.

Proof: From the definition of the distributed measurement $\xi_{i,j}$ in (4.3) and the equality constraint (4.6), one obtains that

$$\begin{aligned}
Q\dot{\xi}_{i,j} &= Q\left(\sum_{k\in\mathcal{N}_j} a_{j,k}(\dot{\mathbf{y}}_{i,k} - \dot{\mathbf{y}}_{i,j}) + d_j(\dot{\mathbf{y}}_d - \dot{\mathbf{y}}_{i,j})\right)\\
&= \sum_{k\in\mathcal{N}_j} a_{j,k}(Q\dot{\mathbf{y}}_{i,k} - Q\dot{\mathbf{y}}_{i,j}) + d_j(Q\dot{\mathbf{y}}_d - Q\dot{\mathbf{y}}_{i,j})\\
&= \sum_{k\in\mathcal{N}_j} a_{j,k}(Q\dot{\mathbf{y}}_d - Q\dot{\mathbf{y}}_{i,j} - Q\dot{\mathbf{y}}_d + Q\dot{\mathbf{y}}_{i,k}) + d_j(Q\dot{\mathbf{y}}_d - Q\dot{\mathbf{y}}_{i,j})\\
&= \sum_{k\in\mathcal{N}_j} a_{j,k}(Q\dot{\hat{\mathbf{y}}}_{v,j} - Q\dot{\mathbf{y}}_{i,j} - Q\dot{\hat{\mathbf{y}}}_{v,k} + Q\dot{\mathbf{y}}_{i,k}) + d_j(Q\dot{\hat{\mathbf{y}}}_{v,j} - Q\dot{\mathbf{y}}_{i,j}).
\end{aligned} \tag{4.7}$$

Define the virtual tracking error $\varepsilon_{i,j} = \hat{\mathbf{y}}_{v,j} - \mathbf{y}_{i,j}$, then (4.7) can be simplified as

$$Q\dot{\xi}_{i,j} = Q\left(\sum_{k\in\mathcal{N}_j} a_{j,k}(\dot{\varepsilon}_{i,j} - \dot{\varepsilon}_{i,k}) + d_j\dot{\varepsilon}_{i,j}\right). \tag{4.8}$$

Define the following notations $\delta\mathbf{u}_{i,j} = \hat{\mathbf{u}}_{v,j} - \mathbf{u}_{i,j}$ and $\delta\mathbf{x}_{i,j} = \hat{\mathbf{x}}_{v,j} - \mathbf{x}_{i,j}$. Therefore, from (4.4) and (4.8) we have

$$\delta\mathbf{u}_{i+1,j} = \delta\mathbf{u}_{i,j} - Q\left(\sum_{k\in\mathcal{N}_j} a_{j,k}(\dot{\varepsilon}_{i,j} - \dot{\varepsilon}_{i,k}) + d_j\dot{\varepsilon}_{i,j}\right). \tag{4.9}$$

Let $\delta\mathbf{u}_i$ and ε_i be the column stack vectors of $\delta\mathbf{u}_{i,j}$ and $\varepsilon_{i,j}$, thus, (4.9) can be written in the following compact form,

$$\delta\mathbf{u}_{i+1} = \delta\mathbf{u}_i - (H\otimes Q)\dot{\varepsilon}_i. \tag{4.10}$$

Taking derivative from both sides of (4.5), and subtracting (4.1) yield

$$\delta\dot{\mathbf{x}}_{i,j} = A\delta\mathbf{x}_{i,j} + B\delta\mathbf{u}_{i,j}. \tag{4.11}$$

Rewriting (4.11) in the compact form yields

$$\delta\dot{\mathbf{x}}_i = (I_N\otimes A)\delta\mathbf{x}_i + (I_N\otimes B)\delta\mathbf{u}_i, \tag{4.12}$$

where $\delta\mathbf{x}_i$ is the column stack vector of $\delta\mathbf{x}_{i,j}$.

Note that $\varepsilon_i = (I_N\otimes C)\delta\mathbf{x}_i$. Substituting (4.12) to (4.10) yields

$$\begin{aligned} & \delta\mathbf{u}_{i+1} \\ = \; & \delta\mathbf{u}_i - (H\otimes Q)(I_N\otimes C)((I_N\otimes A)\delta\mathbf{x}_i + (I_N\otimes B)\delta\mathbf{u}_i) \\ = \; & (I_{mN} - H\otimes QCB)\delta\mathbf{u}_i - (H\otimes QCA)\delta\mathbf{x}_i. \end{aligned} \tag{4.13}$$

Taking λ-norm operation on (4.13) yields

$$\|\delta\mathbf{u}_{i+1}\|_\lambda \leq \mu\|\delta\mathbf{u}_i\|_\lambda + b_1\|\delta\mathbf{x}_i\|_\lambda, \tag{4.14}$$

where $b_1 = |H\otimes QCA|$.

It can be shown that $\hat{\mathbf{x}}_{v,j}(0) = \mathbf{x}_{i,j}(0)$ from (4.5), that is $\delta\mathbf{x}_{i,j}(0) = 0$. Therefore, solving $\delta\mathbf{x}_{i,j}$ from (4.12) we have

$$\delta\mathbf{x}_i = \int_0^t e^{(I_N \otimes A)(t-\tau)} B\delta\mathbf{u}_i(\tau)\,d\tau. \tag{4.15}$$

Taking any generic norm on both sides of (4.15), we have

$$\begin{aligned}
|\delta\mathbf{x}_i| &\leq |B|\int_0^t e^{|I_N \otimes A|(t-\tau)}|\delta\mathbf{u}_i(\tau)|\,d\tau \\
e^{-\lambda t}|\delta\mathbf{x}_i| &\leq e^{-\lambda t}|B|\int_0^t e^{|I_N \otimes A|(t-\tau)}|\delta\mathbf{u}_i(\tau)|\,d\tau \\
e^{-\lambda t}|\delta\mathbf{x}_i| &\leq |B|\int_0^t e^{-(\lambda-|I_N \otimes A|)(t-\tau)}e^{-\lambda\tau}|\delta\mathbf{u}_i(\tau)|\,d\tau.
\end{aligned} \tag{4.16}$$

Again taking λ-norm on (4.16) yields

$$\|\delta\mathbf{x}_i\|_\lambda \leq \frac{|B|}{\lambda - a}|\delta\mathbf{u}_i|_\lambda, \tag{4.17}$$

where $a = |I_N \otimes A|$. Substituting (4.17) to (4.14), we have

$$\|\delta\mathbf{u}_{i+1}\|_\lambda \leq \left(\mu + \frac{b_1|B|}{\lambda - a}\right)\|\delta\mathbf{u}_i\|_\lambda = \tilde{\mu}\|\delta\mathbf{u}_i\|_\lambda. \tag{4.18}$$

If $\lambda > \frac{b_1|B|}{\mu} + a$, then $\tilde{\mu} < 1$. Therefore, $\|\delta\mathbf{u}_i\|_\lambda$ converges to zero as the iteration number increases, i.e., $\mathbf{u}_{i,j} \to \hat{\mathbf{u}}_{v,j}$ and $\mathbf{y}_{i,j} \to \hat{\mathbf{y}}_{v,j}$. ■

By using the graph dependent matrix norm methods in Chapter 2, we can convert the norm inequality convergence condition in Theorem 4.1 to the spectral radius condition. Therefore, we have the following corollary.

Corollary 4.1 *Consider the multi-agent systems (4.1), under Assumptions 4.1, 4.2, the communication graph $\overline{\mathscr{G}}$, and distributed D-type updating rule (4.4). If the learning gain Q is chosen such that*

$$\rho(I_{mN} - H \otimes QCB) < 1,$$

where $\rho(\cdot)$ denotes the spectral radius of a matrix, then the control input $\mathbf{u}_{i,j}$ and output $\mathbf{y}_{i,j}$ converge to $\hat{\mathbf{u}}_{v,j}$ and $\hat{\mathbf{y}}_{v,j}$ respectively as iteration goes to infinity.

As the spectral radius of any given matrix is infimum of any matrix norm (Horn and Johnson, 1985), the spectral radius inequality in Corollary 4.1 is indeed more general than the norm inequality condition in Theorem 4.1.

To satisfy either the norm inequality condition or the spectral radius condition, the communication topology plays an important role for multi-agent coordination to be realizable. When the communication assumption is not strong enough, the coordination goal may not be achievable. For example, if there is an isolated agent or cluster of agents, to which the leader's information cannot be relayed, it is impossible for the multi-agent system to achieve consensus tracking in general. The following Lemma reveals a very useful algebraic property of the communication graph.

Lemma 4.1 *(Ren and Beard, 2008) If the communication graph $\overline{\mathscr{G}}$ contains a spanning tree with the leader being the root, then all the eigenvalues of matrix H have positive real parts.*

With the help of Lemma 1, we have the following sufficient communication requirement for the coordination problem.

Lemma 4.2 *If the communication graph $\overline{\mathscr{G}}$ contains a spanning tree with the leader being the root, there always exists a learning gain Q such that the convergence condition in Corollary 4.1 holds, i.e.,*

$$\rho(I_{mN} - H \otimes QCB) < 1.$$

Proof: Due to Assumption 4.1 that CB is of full column rank, let Q be the pseudoinverse of CB times a positive scalar gain $q > 0$, i.e., $Q = q\left((CB^T)CB\right)^{-1}(CB)^T$. Denote $M = I_{mN} - H \otimes QCB$, hence,

$$M = I_{mN} - qH \otimes I_m.$$

Let $\sigma(H) = \{\lambda_1, \lambda_2, \ldots, \lambda_N\}$ be the spectrum of H. Based on Lemma 4.1, all the eigenvalues have positive real parts. The spectrum of $I_N - qH$ is $\sigma(I_N - qH) = \{1 - q\lambda_1, 1 - q\lambda_2, \ldots, 1 - q\lambda_N\}$. Therefore, choose q sufficiently small such that all the magnitudes of $\sigma(I_N - qH)$ are strictly less than 1. Note the property of Kronecker product, the spectrum of M is the same as $\sigma(I_N - qH)$ except that each eigenvalue has multiplicity of m. Hence, $\rho(M) < 1$. ■

The following result will be used to derive the existence and uniqueness of $\hat{\mathbf{u}}_{v,j}$ and $\hat{\mathbf{y}}_{v,j}$.

Lemma 4.3 *If $\rho(I_{mN} - H \otimes QCB) < 1$ holds, then QCB is nonsingular.*

Proof: It can be shown by contradiction. Assume QCB is singular, that is, QCB has at least one eigenvalue equal to zero. Therefore, $I_{mN} - H \otimes QCB$ has at least one eigenvalue equal to one. Subsequently, we can obtain that $\rho(I_{mN} - H \otimes QCB) \geq 1$, which contradicts to the hypothesis. This completes the proof. ■

The results in Theorem 4.1 and Corollary 4.1 guarantee that the input and output trajectories of each individual agent converge to the corresponding virtual dynamics. It is appropriate time to investigate the properties of the virtual dynamics.

Theorem 4.2 *If the D-type rule (4.4) converges, there exist unique $\hat{\mathbf{u}}_{v,j}$ and $\hat{\mathbf{y}}_{v,j}$ satisfying the virtual dynamics (4.5) and (4.6), specifically,*

$$\hat{\mathbf{u}}_{v,j} = \mathbf{u}_d + (QCB)^{-1} QCAe^{F_D t}(\mathbf{x}_d(0) - \mathbf{x}_{i,j}(0)),$$

and

$$\hat{\mathbf{y}}_{v,j} = \mathbf{y}_d - Ce^{F_D t}(\mathbf{x}_d(0) - \mathbf{x}_{i,j}(0)),$$

where $F_D = (I - B(QCB)^{-1}QC)A$.

Proof: From (4.6), the dynamics (4.5), and (4.2), we have

$$\begin{aligned} Q\dot{\mathbf{y}}_d &= Q\dot{\hat{\mathbf{y}}}_{v,j} \\ QCA\mathbf{x}_d + QCB\mathbf{u}_d &= QCA\hat{\mathbf{x}}_{v,j} + QCB\hat{\mathbf{u}}_{v,j} \\ QCB(\mathbf{u}_d - \hat{\mathbf{u}}_{v,j}) &= -QCA(\mathbf{x}_d - \hat{\mathbf{x}}_{v,j}) \end{aligned} \tag{4.19}$$

Define $\delta\hat{\mathbf{u}}_{v,j} = \mathbf{u}_d - \hat{\mathbf{u}}_{v,j}$, and $\delta\hat{\mathbf{x}}_{v,j} = \mathbf{x}_d - \hat{\mathbf{x}}_{v,j}$. Note that QCB is nonsingular, (4.19) can be written as

$$\delta\hat{\mathbf{u}}_{v,j} = -(QCB)^{-1}QCA\delta\hat{\mathbf{x}}_{v,j}. \tag{4.20}$$

Taking derivative on both sides of (4.5), and subtracting it from (4.2) yield

$$\delta\dot{\hat{\mathbf{x}}}_{v,j} = A\delta\hat{\mathbf{x}}_{v,j} + B\delta\hat{\mathbf{u}}_{v,j}. \tag{4.21}$$

With initial condition $\delta\hat{\mathbf{x}}_{v,j}(0) = \mathbf{x}_d(0) - \mathbf{x}_{i,j}(0)$, substituting (4.20) to (4.21) yields

$$\delta\dot{\hat{\mathbf{x}}}_{v,j} = (I_n - B(QCB)^{-1}QC)A\delta\hat{\mathbf{x}}_{v,j}. \tag{4.22}$$

As (4.22) is a linear differential equation, its solution exists and is unique. Therefore, both $\hat{\mathbf{u}}_{v,j}$ and $\hat{\mathbf{y}}_{v,j}$ exist and are unique.

Denote $F_D = (I_n - B(QCB)^{-1}QC)A$. Solve $\delta\hat{\mathbf{x}}_{v,j}$ from (4.22), we have

$$\delta\hat{\mathbf{x}}_{v,j} = e^{F_D t}(\mathbf{x}_d(0) - \mathbf{x}_{i,j}(0)).$$

Therefore, from (4.20) we have

$$\hat{\mathbf{u}}_{v,j} = \mathbf{u}_d + (QCB)^{-1}QCAe^{F_D t}(\mathbf{x}_d(0) - \mathbf{x}_{i,j}(0)),$$

and

$$\hat{\mathbf{y}}_{v,j} = \mathbf{y}_d - Ce^{F_D t}(\mathbf{x}_d(0) - \mathbf{x}_{i,j}(0)).$$

■

The learning gain Q is designed independent of A. It is interesting to note that the system matrix A does not affect the convergence property of ILC algorithm. However, Theorem 4.2 says when there is a discrepancy between initial condition and the desired one, the final output trajectory is decided by all the system parameters. In the D-type ILC rule, we only have the freedom to tune one parameter, that is the learning gain Q. It is hard to ensure both the ILC convergence condition and minimizing the effect of $e^{F_D t}$. This motivates us to consider PD-type updating rule in the next section, which gives us two degree of freedom to ensure both convergence and final performance.

Corollary 4.2 *If Q is of full column rank, then*

$$\hat{\mathbf{y}}_{v,j} = \mathbf{y}_d - C(\mathbf{x}_d(0) - \mathbf{x}_{i,j}(0)).$$

Proof: If Q is of full column rank, it can be shown that $\dot{\mathbf{y}}_d - \dot{\hat{\mathbf{y}}}_{v,j}$ has to be zero in order to satisfy the equality constraint (4.6). Hence, by integration we have

$$\begin{aligned}
\mathbf{y}_d(t) - \mathbf{y}_d(0) &= \hat{\mathbf{y}}_{v,j}(t) - \hat{\mathbf{y}}_{v,j}(0) \\
\hat{\mathbf{y}}_{v,j}(t) &= \mathbf{y}_d(t) - \mathbf{y}_d(0) + \hat{\mathbf{y}}_{v,j}(0) \\
\hat{\mathbf{y}}_{v,j}(t) &= \mathbf{y}_d(t) - C(\mathbf{x}_d(0) - \mathbf{x}_{i,j}(0)).
\end{aligned}$$

■

Corollary 4.2 implies that when Q is of full column rank or just nonsingular, the final output trajectory of each follower is identical to the leader trajectory with a constant shift, and the discrepancy is simply the initial output difference $C(\mathbf{x}_d(0) - \mathbf{x}_{i,j}(0))$.

4.3.2 Distributed PD-type Updating Rule

Motivated by the previous section that D-type ILC rule has only one degree of freedom which ensures convergence. To improve the final performance as the iteration number gets large, we can apply the PD-type updating rule. Let the distributed measurement be the same as the one in the D-type case. The proposed PD-type updating rule for the jth agent is

$$\mathbf{u}_{i+1,j} = \mathbf{u}_{i,j} + Q(\dot{\xi}_{i,j} + R\xi_{i,j}), \tag{4.23}$$

where R is another learning gain to be designed.

Comparing (4.23) with (4.4), PD-type rule has one extra term $R\xi_{i,j}$ compared with the D-type rule. It will be shown that it is the extra term which enables us to tune the final performance.

Theorem 4.3 *Consider the multi-agent systems (4.1), under Assumptions 4.1, 4.2, the communication graph $\overline{\mathscr{G}}$, and distributed PD-type updating rule (4.23). If the learning gain Q is chosen such that*

$$\rho(I_{mN} - H \otimes QCB) < 1,$$

then the ILC rule is stable and output trajectory of any follower converges, in particular as the iteration number increases to infinity

$$\mathbf{u}_{i,j} = \mathbf{u}_d + (QCB)^{-1}Q(CA + RC)e^{F_{PD}t}(\mathbf{x}_d(0) - \mathbf{x}_{i,j}(0)),$$

and

$$\mathbf{y}_{i,j} = \mathbf{y}_d - Ce^{F_{PD}t}(\mathbf{x}_d(0) - \mathbf{x}_{i,j}(0)),$$

where $F_{PD} = A - B(QCB)^{-1}Q(CA + RC)$.

The proof of Theorem 4.3 is omitted here as it can be derived analogously to Theorem 4.1. In Theorem 4.3, Q can be tuned to make the ILC rule stable, and R is used to modify the final performance as iteration number gets large. Hence, R should be designed such that F_{PD} is Hurwitz. Then, the final output trajectory of each individual follower will converge exponentially to the leader's trajectory.

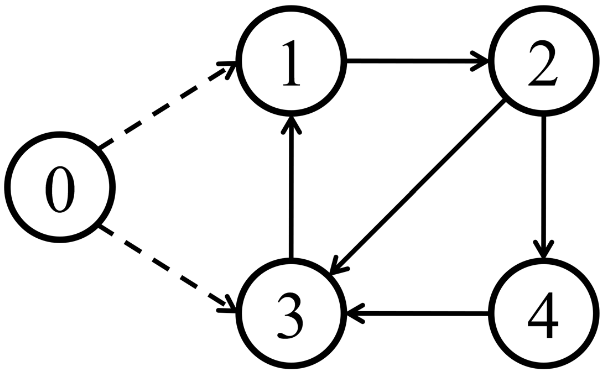

Figure 4.1: Communication topology among agents in the network.

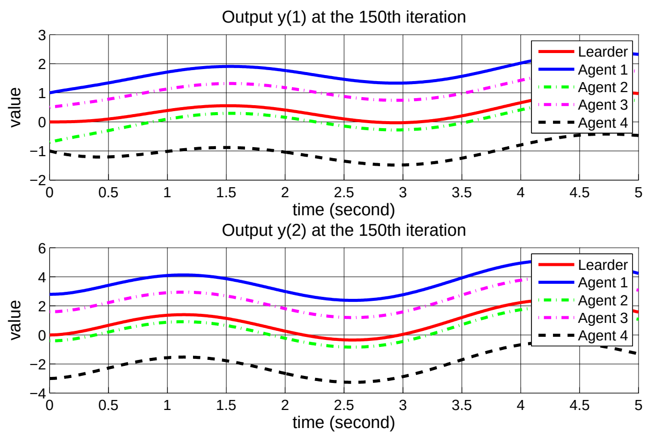

Figure 4.2: Output trajectories at the 150th iteration under D-type ILC learning rule.

Rewrite F_{PD} as

$$F_{PD} = (I_n - B(QCB)^{-1}QC)A + B(QCB)^{-1}Q(-R)C,$$

which can be interpreted as a static output feedback stabilization problem (Syrmos et al., 1997) with system matrix $(I_n - B(QCB)^{-1}QC)A$, input gain $B(QCB)^{-1}Q$, and output matrix C. R is the output feedback gain. R should be chosen such that the output feedback system is stable.

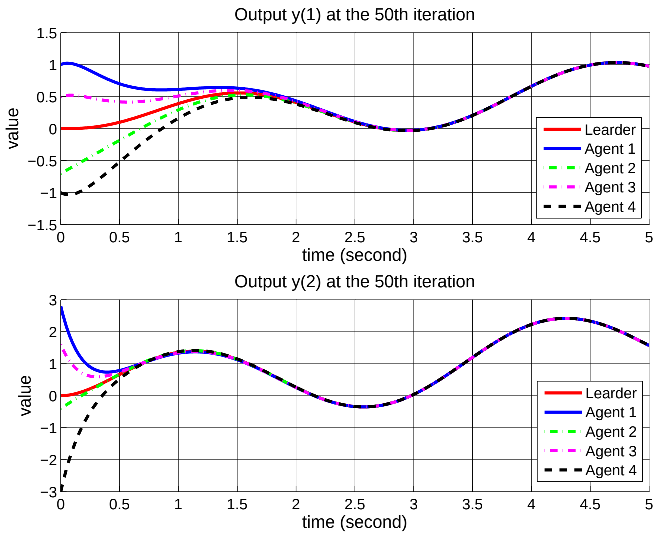

Figure 4.3: Output trajectories at the 50th iteration under PD-type ILC learning rule.

4.4 Illustrative Example

To verify the theoretical results in the previous section, two illustrative examples are presented here. Consider a group of four followers with their dynamics governed by the following model,

$$A = \begin{bmatrix} 0 & 1 \\ -6 & -5 \end{bmatrix}, \quad B = \begin{bmatrix} 0 \\ 1 \end{bmatrix}, \quad C = \begin{bmatrix} 1 & 0 \\ 2 & 1 \end{bmatrix}.$$

The communication among all the agents in the network is depicted in Figure. 4.1. Vertex 0 represents the leader agent. It has two direct edges (dashed lines) to agents 1 and 3, which means that the agents 1 and 3 are able to access to the leader's information. The communication among followers are represented by the solid lines. It is easy to verify that the complete information flow graph $\overline{\mathscr{G}}$ contains a spanning tree with the leader being its root. From Figure. 4.1, we can write down the graph Laplacian for the

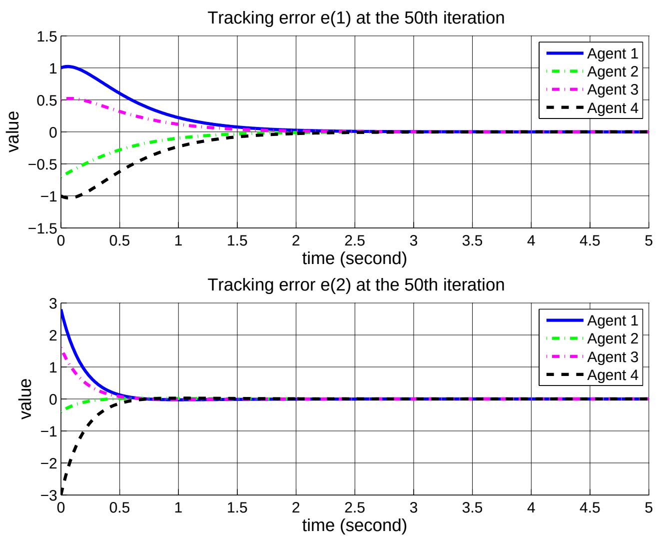

Figure 4.4: Tracking error profiles at the 50th iteration under PD-type ILC learning rule.

followers below,

$$L = \begin{bmatrix} 1 & 0 & -1 & 0 \\ -1 & 1 & 0 & 0 \\ 0 & -1 & 2 & -1 \\ 0 & -1 & 0 & 1 \end{bmatrix},$$

and $D = \mathrm{diag}(1,0,1,0)$ which represents the information flow from leader to followers.

The leader's input is chosen as $u_d = t + 4\sin(2t)$, $t \in [0,5]$, and initial condition $\mathbf{x}_d(0) = [0,\ 0]^T$. The initial conditions for followers are $\mathbf{x}_{i,1}(0) = [1,\ 0.8]^T$, $\mathbf{x}_{i,2}(0) = [-0.7,\ 1]^T$, $\mathbf{x}_{i,3}(0) = [0.5,\ 0.6]^T$, and $\mathbf{x}_{i,4}(0) = [-1,\ -1]^T$. Obviously, initial state errors are nonzero. Note that only the output is accessible to agents 1 and 3. We apply the updating rules (4.4) and (4.23) with learning gains $Q = [0.0667,\ 0.333]$ and $R = \mathrm{diag}(5,5)$. Check the convergence condition,

$$\rho(I_4 - H \otimes QCB) = 0.7715 < 1,$$

which satisfies the convergence requirement in Corollary 4.1. Figure. 4.2 shows the output profiles of all agents at the 150th iteration under D-type updating rule. The followers' output can track the general trend of the leader, but large deviations exist. Simple calculation shows that the spectrum of F_D is $\sigma(F_D) = \{0,\ -2.2\}$. It means that one eigenmode converges to zero exponentially, the other one is a constant. It implies that the discrepancies between leader's trajectory and followers' approach to a constant when time is large. The theoretical prediction perfectly matches the observation in simulation. Figures. 4.3 and 4.4 describe the trajectory and error profiles under the PD-type rule. It can be seen that the tracking error converges exponentially to zero. Calculate the spectrum of F_{PD}, we have $\sigma(F_{PD}) = \{-2.2,\ -5\}$, which are all stable. This also matches the actual simulation.

4.5 Conclusion

This chapter investigates the initial state error problem in the multi-agent setup, and each agent is able to be reset to a fixed initial position. Such an assumption is less restrictive than the perfect initial condition. It has been shown under the imperfect initial condition, the D-type learning rule is still convergent. However, the final trajectory has deviations from the leader's. To improve the performance, PD-type updating rule is proposed, which gives the designer more freedom to tune the final control performance. Numerical examples verify the obtained results. The obtained results can be easily generalized to PID-type learning rules.

Chapter 5

P-type Iterative Learning for Non-parameterized Systems with Uncertain Local Lipschitz Terms

5.1 Background

Traditional ILC has two fundamental postulates, namely system repeatability and global Lipschitz condition (GLC). The system repeatability consists of three conditions over the iteration axis: the identical initialization condition ($i.i.c.$), identical system dynamics, and identical control task. Relaxation or removal of each repeatability condition leads to the great advances in recent ILC research. Much effort has been devoted to relaxing or even removing the $i.i.c.$ (Xu and Qu, 1998; Chen et al., 1999; Sun and Wang, 2002; Fang and Chow, 2003; Park, 2005; Xu and Yan, 2005). For detailed discussion on initial condition problem, please refer to Chapter 4. Relaxation or removal of identical system dynamics and identical control task have relatively few results available. The results in Saab (1994) show that if the fluctuations in system dynamics can be modeled by bounded term, uniform convergence can be obtained. Chen et al. (1997) show that if the variation of reference trajectories between two consecutive iterations is bounded, the tracking error will converge to some bounded ball. Liu et al. (2010) and Yin et al. (2010) successfully apply ILC learning rule to iteration-varying control tasks and iteration-varying dynamic systems by introducing a high-order internal model to the controller design.

Unlike the rich literature on system repeatability issues, it seems impossible to remove the GLC in the contraction-mapping (CM) based ILC. Up to date, it is not clear whether the CM based ILC is applicable to local Lipschitz systems or not. Composite energy function (CEF) based ILC (Xu and Tan, 2002a) is able to handle local Lipschitz systems. However, system dynamics must be in linear in parameter form, and full state information must be used for feedback or nonlinear compensation. When dealing with real-life control problems we always consider CM based ILC in the first place because of the following advantages: extremely simple algorithms, little knowledge requirement of state dynamics, exponential convergence rate, and applicable to nonaffine-in-input processes. Thus, it is very suitable and highly desired for practical control problems. The major disadvantage is due to the requirement of GLC. Traditional CM based ILC, is essentially a feedforward control, and only output information is used. There is no feedback control for state dynamics. GLC is required for the drift term in the unforced system, such that in the worst case when the unforced system is unstable, the finite escape time phenomenon will not occur. From the practical point of view, most control engineering systems are inherently stable, for instance the mechanical systems with friction, thermal systems with heat dissipation, and electrical systems with inner resistance. In traditional CM based ILC analysis, the drift term is always treated as an unstable term by taking norm operation, even though the drift term could be originally a stabilizing term. This limitation is innate in CM based analysis. To overcome this limitation, in this chapter we introduce Lyapunov method into the CM based ILC analysis, aiming at fully making use of the inherent stability of the unforced system. In the sequel, ILC can be applied to more generic systems with local Lipschitz nonlinearities. Our objective is to retain the CM based ILC, meanwhile incorporate Lyapunov method to facilitate the ILC convergence analysis, hence widen the applicability of CM based ILC to more generic dynamic systems with nonlinearities beyond GLC.

The extension of ILC approach to local Lipschitz systems has great positive impact on the multi-agent coordination problems. In the previous three chapters, all the agent models are assumed to be global Lipschitz continuous. By using the results in this chapter, it is possible to design distributed iterative learning rules for certain classes of local Lipschitz system models.

The rest of this chapter is organized as follows. The motivation and problem formulation are described in Section 5.2. Sections 5.3 and 5.4 present the main results. Specifically, Section 5.3.1 introduces some preliminary results that will be used to prove the main theorems. Section 5.3.2 develops the Lyapunov function based sufficient convergence conditions. Section 5.3.3 studies the local Lipschitz system whose unstable part is global Lipschitz continuous. Section 5.4.1 studies the system with bounded drift term. Section 5.4.2 investigates the class of bounded energy bounded state systems subject to control saturation. Finally, conclusions are drawn in Section 5.5.

5.2 Motivation and Problem Description

This section discusses the motivation and problem description.

5.2.1 Motivation

ILC is a partially model-free control as the controller design does not require the detailed information on system model. To illustrate the motivation of the research in this work, consider the following first-order dynamics,

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

where $f(x)$ is the drift term that determines the stability of the unforced system. $f(x)$ may contain both stable and unstable terms. In traditional ILC analysis, norm operations are taken on the system dynamics in order to find out the relation between the magnitudes of system state and control input. As a result, both the stable and unstable terms in $f(x)$ are treated as if they were unstable. As such, the bounding function for system state is too conservative, which makes the analysis method only applicable to the system whose $f(x)$ satisfies GLC. This restricts the application of ILC to a very limited class of systems.

Up to date, it is not clear how the local Lipschitz terms in $f(x)$ affect the convergence property of ILC rule. However, it has been observed that many local Lipschitz systems demonstrate convergence property under ILC in simulation studies. This motivates us to investigate to what kind of local Lipschitz system the ILC rule is applicable. Three classes of local Lipschitz systems are studied in this chapter. First, we distinguish the stable and unstable terms in the system dynamics, and develop several sufficient conditions in the form of Lyapunov function based criteria. Next, we study the bounded local Lipschitz terms. Finally, the convergence property of the uniformly bounded energy bounded state system under control saturation is analyzed.

5.2.2 Problem Description

Consider the system model in (5.1),

$$\dot{\mathbf{x}}(t) = \mathbf{f}(\mathbf{x}(t)) + \mathbf{g}(\mathbf{x}(t))\mathbf{u}(t), \tag{5.1a}$$

$$\mathbf{y}(t) = \mathbf{h}(\mathbf{x}(t), \mathbf{u}(t)), \tag{5.1b}$$

where $\mathbf{x} \in \mathbb{R}^n$, $\mathbf{u} \in \mathbb{R}^p$, and $\mathbf{y} \in \mathbb{R}^m$ are the system state, input and output vectors, $\mathbf{f}(\mathbf{x})$ belongs to a class of local Lipschitz functions, $\mathbf{g}(\mathbf{x})$ is a continuously differentiable and bounded function for all $\mathbf{x} \in \mathbb{R}^n$, and $\mathbf{h}(\mathbf{x}, \mathbf{u})$ is a differentiable function. Furthermore, $\frac{\partial \mathbf{h}}{\partial \mathbf{u}}(\mathbf{x}, \mathbf{u})$ is of full rank, and both $\frac{\partial \mathbf{h}}{\partial \mathbf{u}}(\mathbf{x}, \mathbf{u})$ and $\frac{\partial \mathbf{h}}{\partial \mathbf{x}}(\mathbf{x}, \mathbf{u})$ are bounded for all $\mathbf{x} \in \mathbb{R}^n$ and $\mathbf{u} \in \mathbb{R}^p$. For notional simplicity, the following three functional norms are defined,

$$\begin{aligned} \|\mathbf{g}(\mathbf{x})\| &= \sup_{\mathbf{x}\in\mathbb{R}^n} |\mathbf{g}(\mathbf{x})|, \\ \left\|\frac{\partial \mathbf{h}}{\partial \mathbf{u}}(\mathbf{x}, \mathbf{u})\right\| &= \sup_{\mathbf{x}\in\mathbb{R}^n, \mathbf{u}\in\mathbb{R}^p} \left|\frac{\partial \mathbf{h}}{\partial \mathbf{u}}(\mathbf{x}, \mathbf{u})\right|, \\ \left\|\frac{\partial \mathbf{h}}{\partial \mathbf{x}}(\mathbf{x}, \mathbf{u})\right\| &= \sup_{\mathbf{x}\in\mathbb{R}^n, \mathbf{u}\in\mathbb{R}^p} \left|\frac{\partial \mathbf{h}}{\partial \mathbf{x}}(\mathbf{x}, \mathbf{u})\right|. \end{aligned}$$

The desired trajectory is $\mathbf{y}_d \in \mathscr{C}^1[0, T]$. For any given $\mathbf{y}_d$ to be realizable, it is assumed that there exist $\mathbf{x}_d \in \mathscr{C}^1[0, T]$ and $\mathbf{u}_d \in \mathscr{C}[0, T]$ such that the following dynamics hold.

$$\dot{\mathbf{x}}_d = \mathbf{f}(\mathbf{x}_d) + \mathbf{g}(\mathbf{x}_d)\mathbf{u}_d, \tag{5.2a}$$

$$\mathbf{y}_d = \mathbf{h}(\mathbf{x}_d, \mathbf{u}_d). \tag{5.2b}$$

Let ω_i denote a variable at the ith iteration, where $\omega \in \{\mathbf{x}, \mathbf{y}, \mathbf{u}\}$. Define the tracking error at the ith iteration as $\mathbf{e}_i = \mathbf{y}_d - \mathbf{y}_i$. The traditional P-type ILC law is constructed below

$$\mathbf{u}_{i+1} = \mathbf{u}_i + \Gamma \mathbf{e}_i, \tag{5.3}$$

where Γ is some suitable learning gain such that

$$\left\|I_m - \Gamma\frac{\partial \mathbf{h}}{\partial \mathbf{u}}\right\| = \sup_{\mathbf{x}\in\mathbb{R}^n, \mathbf{u}\in\mathbb{R}^p}\left|I_m - \Gamma\frac{\partial \mathbf{h}}{\partial \mathbf{u}}\right| \leq \rho < 1, \tag{5.4}$$

I_m is the identity matrix, and the subscript m denotes its dimension.

We say the P-type learning rule is convergent if $\mathbf{u}_i$ converges to $\mathbf{u}_d$ as the learning iteration approaches to infinity. Subsequently, the tracking error $\mathbf{e}_i$ converges to zero as well. It is a well established result that (5.3) is convergent if $\mathbf{f}(\mathbf{x})$ is global Lipschitz continuous in $\mathbf{x}$ (Xu and Tan, 2003). The problem in this chapter is to explore to what extent the learning rule (5.3) can be applied to local Lipschitz systems.

To restrict our discussion, the following assumptions are imposed.

Assumption 5.1 *Let $\mathscr{D}$ be a compact subset of $\mathbb{R}^n$. For any $\mathbf{z}_1$, $\mathbf{z}_2 \in \mathscr{D}$, there exists a continuous bounding function $\phi(\mathbf{z})$ such that*

$$|\mathbf{f}(\mathbf{z}_1) - \mathbf{f}(\mathbf{z}_2)| \leq \phi(\mathbf{z})|\mathbf{z}_1 - \mathbf{z}_2|,$$

where $\mathbf{z} = [\mathbf{z}_1^T, \mathbf{z}_2^T]^T$. Furthermore, if $\mathscr{D} \to \mathbb{R}^n$, then $\lim_{|\mathbf{z}|\to\infty} \phi(\mathbf{z}) \to \infty$.

Remark 5.1 *If $\phi(\mathbf{z})$ has an upper bound for all $\mathbf{z} \in \mathbb{R}^{2n}$, Assumption 5.1 degenerates to the global Lipschitz condition. Thus, traditional λ-norm analysis can be applied to prove ILC convergence. As $\phi(\mathbf{z})$ is unbounded in $\mathbb{R}^{2n}$, Assumption 5.1 represents a class of local Lipschitz functions, e.g., high-order polynomials. Under Assumption 5.1, if the system state $\mathbf{x}_i$ is bounded along the iteration axis when controller (5.3) is applied, then the local Lipschitz condition can be treated as global one in the learning context. This observation motivates the two-step idea to prove ILC convergence. First, show that the state is confined in certain compact set $\mathscr{D}$ under learning rule (5.3) by taking advantages of system properties. Next, $\phi(\mathbf{z})$ is bounded on $\mathscr{D}$ as it is a continuous function. Therefore, λ-norm can be utilized to construct a contraction-mapping.*

Assumption 5.2 *The initial state is reset to the desired initial state at every iteration, i.e.,* $\mathbf{x}_i(0) = \mathbf{x}_d(0)$.

Assumption 5.2 is one of the most commonly used assumptions in ILC literature. For simplicity we assume the perfect resetting condition since the initial condition problem is not the main topic in this chapter. As discussed in Chapters 2 and 4, there are many alternatives to relax such a restrictive condition.

5.3 Convergence Properties with Lyapunov Stability Conditions

In this section, we investigate several classes of local Lipschitz systems with the help of Lyapunov analysis method, and study their P-type ILC convergence.

5.3.1 Preliminary Results

In this subsection, two useful lemmas are introduced. They will be utilized to prove the main results in this chapter.

Lemma 5.1 *If* $\alpha \in \mathscr{K}_\infty$[1] *and convex, then for any* $r > 0$ *there exist some positive constants* k *and* b *such that the following inequality holds*

$$\alpha^{-1}(r) \leq kr + b.$$

In Lemma 5.1, $\alpha \in \mathscr{K}_\infty$ implies α is a one-to-one mapping from $[0, \infty)$ to $[0, \infty)$. Hence, the inverse function α^{-1} exists. As α is a convex function, α^{-1} has to be concave. Therefore, we can find a linear bounding function for α^{-1}. The constants k and b

[1] $\mathscr{K}$ is a class of functions $[0, \infty) \to [0, \infty)$ which are zeros at zero, strictly increasing, and continuous. $\mathscr{K}_\infty$ is a subset of $\mathscr{K}$ whose elements are unbounded (Khalil, 2002).

depend on α. Lemma 5.1 will be used to establish the growth rate of system state as a function of input.

Lemma 5.2 *Consider system (5.1) under the P-type learning rule (5.3), Assumptions 5.1 and 5.2. If the system state satisfies the following inequality*

$$|\mathbf{x}(t)| < l \int_0^t |\mathbf{u}(\tau)|\, d\tau + \eta(t), \tag{5.5}$$

where l is a positive constant and $\eta(t)$ is a bounded continuous function of time, then the P-type learning rule (5.3) is convergent.

Proof: Substituting (5.1b) into the P-type learning rule (5.3) yields

$$\begin{aligned}
\mathbf{u}_{i+1} &= \mathbf{u}_i + \Gamma \mathbf{e}_i \\
&= \mathbf{u}_i + \Gamma(\mathbf{h}(\mathbf{x}_d, \mathbf{u}_d) - \mathbf{h}(\mathbf{x}_i, \mathbf{u}_i)) \\
&= \mathbf{u}_i + \Gamma(\mathbf{h}(\mathbf{x}_d, \mathbf{u}_d) - \mathbf{h}(\mathbf{x}_d, \mathbf{u}_i)) + \Gamma(\mathbf{h}(\mathbf{x}_d, \mathbf{u}_i) - \mathbf{h}(\mathbf{x}_i, \mathbf{u}_i)).
\end{aligned} \tag{5.6}$$

By using the Mean Value Theorem (Khalil, 2002), the last two terms in (5.6) can be respectively written as

$$\begin{aligned}
\mathbf{h}(\mathbf{x}_d, \mathbf{u}_d) - \mathbf{h}(\mathbf{x}_d, \mathbf{u}_i) &= \frac{\partial \mathbf{h}}{\partial \mathbf{u}}(\mathbf{x}_d, \hat{\mathbf{u}}_i)(\mathbf{u}_d - \mathbf{u}_i), \\
\mathbf{h}(\mathbf{x}_d, \mathbf{u}_i) - \mathbf{h}(\mathbf{x}_i, \mathbf{u}_i) &= \frac{\partial \mathbf{h}}{\partial \mathbf{x}}(\hat{\mathbf{x}}_i, \mathbf{u}_i)(\mathbf{x}_d - \mathbf{x}_i),
\end{aligned}$$

where $\hat{\mathbf{u}}_i = \mathbf{u}_d + \theta_i^{\mathbf{u}}(\mathbf{u}_i - \mathbf{u}_d)$, $\hat{\mathbf{x}}_i = \mathbf{x}_d + \theta_i^{\mathbf{x}}(\mathbf{x}_i - \mathbf{x}_d)$, and $\theta_i^{\mathbf{u}}$, $\theta_i^{\mathbf{x}} \in [0,\, 1]$. Thus, (5.6) can be simplified as

$$\mathbf{u}_{i+1} = \left(I_m - \Gamma \frac{\partial \mathbf{h}}{\partial \mathbf{u}}\right)\mathbf{u}_i - \Gamma \frac{\partial \mathbf{h}}{\partial \mathbf{x}}\mathbf{x}_i + \Gamma \frac{\partial \mathbf{h}}{\partial \mathbf{u}}\mathbf{u}_d + \Gamma \frac{\partial \mathbf{h}}{\partial \mathbf{x}}\mathbf{x}_d. \tag{5.7}$$

Since $\frac{\partial \mathbf{h}}{\partial \mathbf{u}}$ and $\frac{\partial \mathbf{h}}{\partial \mathbf{x}}$ are bounded, denote $d = \|\frac{\partial \mathbf{h}}{\partial \mathbf{u}}\|$ and $c = \|\frac{\partial \mathbf{h}}{\partial \mathbf{x}}\|$. Noticing the convergence condition (5.4) and inequality (5.5), taking norm on both sides of (5.7) yields

$$\begin{aligned} |\mathbf{u}_{i+1}| &\leq \left| I_m - \Gamma \frac{\partial \mathbf{h}}{\partial \mathbf{u}} \right| |\mathbf{u}_i| + c|\Gamma| \left(l \int_0^t |\mathbf{u}_i(\tau)|\, d\tau + \eta(t) \right) + d|\Gamma \mathbf{u}_d| + c|\Gamma \mathbf{x}_d| \\ &\leq \rho |\mathbf{u}_i| + cl|\Gamma| \int_0^t |\mathbf{u}_i(\tau)|\, d\tau + \underbrace{c|\Gamma|\eta(t) + d|\Gamma \mathbf{u}_d| + c|\Gamma \mathbf{x}_d|}_{\text{iteration-invariant}}. \qquad (5.8) \end{aligned}$$

Denote the supremum of the iteration-invariant term in (5.8) as $\Delta = c|\Gamma|\|\eta\| + d|\Gamma|\|\mathbf{u}_d\| + c|\Gamma|\|\mathbf{x}_d\|$, which is a constant. Therefore, taking λ-norm on both sides of (5.8) yields

$$\|\mathbf{u}_{i+1}\|_\lambda \leq \left(\rho + \frac{cl|\Gamma|}{\lambda} \right) \|\mathbf{u}_i\|_\lambda + \Delta. \qquad (5.9)$$

Choosing a sufficiently large λ such that $\rho + \frac{cl|\Gamma|}{\lambda} = \bar{\rho} < 1$. Therefore, $\|\mathbf{u}_i\|_\lambda$ is bounded for any iteration i, and

$$\|\mathbf{u}_i\|_\lambda \leq \frac{\Delta}{1 - \bar{\rho}}.$$

From the analysis above, $\|\mathbf{u}_i\|$ must be bounded for all iterations as the supremum norm and λ-norm are equivalent. Based on the inequality (5.5), $\|\mathbf{x}_i\|$ must belong to certain compact set $\mathscr{D}$ for all iterations. Therefore, $\mathbf{f}(\mathbf{x})$ can be treated as a global Lipschitz function in the learning context. Now utilizing Assumptions 5.1 and 5.2, the convergence of P-type learning rule (5.3) can be proved by traditional λ-norm analysis (Xu and Tan, 2003). ■

The condition (5.5) is more general than the global Lipschitz condition in ILC literature. To demonstrate this point, assume that $\mathbf{f}(\mathbf{x})$ satisfies the global Lipschitz condition, that is for any $\mathbf{z}_1, \mathbf{z}_2 \in \mathbb{R}^n$,

$$|\mathbf{f}(\mathbf{z}_1) - \mathbf{f}(\mathbf{z}_2)| \leq L|\mathbf{z}_1 - \mathbf{z}_2|,$$

where L is the global Lipschitz constant.

From (5.1a) the solution of $\mathbf{x}(t)$ can be written as

$$\mathbf{x}(t) = \mathbf{x}(0) + \int_0^t (\mathbf{f}(\mathbf{x}(\tau)) + \mathbf{g}(\mathbf{x}(\tau))\mathbf{u}(\tau))\, d\tau$$

Taking norm on both sides and applying global Lipschitz condition on $\mathbf{f}(\mathbf{x})$ yield

$$
\begin{aligned}
|\mathbf{x}(t)| &\leq |\mathbf{x}(0)| + \int_0^t (|\mathbf{f}(\mathbf{x})| + |\mathbf{g}(\mathbf{x})\mathbf{u}(\tau)|)\, d\tau \\
&\leq |\mathbf{x}(0)| + \int_0^t |\mathbf{f}(0)|\, d\tau + \int_0^t \|\mathbf{g}(\mathbf{x})\||\mathbf{u}(\tau)|\, d\tau + \int_0^t L|\mathbf{x}(\tau)|\, d\tau.
\end{aligned}
$$

Therefore, applying Gronwall-Bellman's Lemma (Khalil, 2002) yields

$$
\begin{aligned}
|\mathbf{x}(t)| &\leq e^{Lt}|\mathbf{x}(0)| + \int_0^t e^{L(t-\tau)}\|\mathbf{g}(\mathbf{x})\||\mathbf{u}(\tau)|\, d\tau + \int_0^t e^{L(t-\tau)}|\mathbf{f}(0)|\, d\tau \\
&\leq l\int_0^t |\mathbf{u}(\tau)|\, d\tau + \eta(t),
\end{aligned}
$$

where $l = \|\mathbf{g}(\mathbf{x})\| \int_0^T e^{L(T-\tau)}\, d\tau$ and $\eta(t) = e^{Lt}|\mathbf{x}(0)| + \int_0^t e^{L(t-\tau)}|\mathbf{f}(0)|\, d\tau$. Therefore, we can conclude that the global Lipschitz condition is a special case of (5.5).

5.3.2 Lyapunov Stable Systems

To explore the applicability of ILC to local Lipschitz systems, the first natural candidate is the Lyapunov stable system. Furthermore, many industry processes are stable due to heat dissipation and frictions. This section focuses on Lyapunov stable systems.

Theorem 5.1 *Consider system (5.1) under the P-type learning rule (5.3), Assumptions 5.1 and 5.2. If there exist $\alpha_1, \alpha_2 \in \mathscr{K}_\infty$, a continuously differentiable function $W(\mathbf{x})$, a positive semidefinite function $\alpha_3(\mathbf{x})$, and a positive constant γ, such that*

$$\alpha_1(|\mathbf{x}|) \leq W(\mathbf{x}) \leq \alpha_2(|\mathbf{x}|) \tag{5.10}$$

$$\frac{\partial W}{\partial \mathbf{x}}(\mathbf{f}(\mathbf{x}) + \mathbf{g}(\mathbf{x})\mathbf{u}) \leq -\alpha_3(\mathbf{x}) + \gamma|\mathbf{u}|. \tag{5.11}$$

Furthermore, if α_1 is convex, then the P-type learning rule (5.3) is convergent.

Proof: From the inequality (5.11), we have

$$
\begin{aligned}
W(\mathbf{x}(t)) &\leq -\int_0^t \alpha_3(\mathbf{x}(\tau))\, d\tau + \gamma\int_0^t |\mathbf{u}(\tau)|\, d\tau + W(\mathbf{x}(0)) \\
&\leq \gamma\int_0^t |\mathbf{u}(\tau)|\, d\tau + W(\mathbf{x}(0)).
\end{aligned} \tag{5.12}
$$

By using the comparison function in (5.10), (5.12) can be written as

$$\alpha_1(|\mathbf{x}|) \leq \gamma \int_0^t |\mathbf{u}(\tau)|\, d\tau + W(\mathbf{x}(0)). \tag{5.13}$$

Notice that α_1 is convex, from Lemma 5.1 we have the following inequality

$$|\mathbf{x}| \leq \gamma k \int_0^t |\mathbf{u}(\tau)|\, d\tau + kW(\mathbf{x}(0)) + b, \tag{5.14}$$

where k and b are some constants. Hence, (5.14) satisfies the condition in (5.5). Therefore, we can conclude that the P-type learning rule is convergent by using Lemma 5.2

■

Take a close look at equations (5.10) and (5.11), when input $\mathbf{u}$ is set to zero, the unforced system of (5.1a) is Lyapunov stable. Theorem 5.1 provides a sufficient condition for P-type rule convergence. Next we will investigate what kind of systems satisfy (5.10) and (5.11).

Lemma 5.3 *If the unforced system of (5.1a) is Lyapunov stable, then there exist a continuously differentiable function $W(\mathbf{x})$, comparison functions $\alpha_1, \alpha_2 \in \mathscr{K}$, and a positive semidefinite function $\alpha_3(\mathbf{x})$, such that*

$$\alpha_1(|\mathbf{x}|) \leq W(\mathbf{x}) \leq \alpha_2(|\mathbf{x}|),$$

$$\tfrac{\partial W}{\partial \mathbf{x}}(\mathbf{f}(\mathbf{x}) + \mathbf{g}(\mathbf{x})\mathbf{u}) \leq -\alpha_3(\mathbf{x}) + \gamma|\mathbf{u}|.$$

Comparing Lemma 5.3 with Theorem 5.1, we can notice that in Lemma 5.3 α_1 and α_2 belong to $\mathscr{K}$ functions instead of $\mathscr{K}_\infty$ functions, and α_1 is not necessarily a convex function.

Proof: As the unforced system of (5.1a) is Lyapunov stable, by Converse Theorem (Khalil, 2002) there exists a Lyapunov function $V(\mathbf{x})$ such that

$$\bar{\alpha}_1(|\mathbf{x}|) \leq V(\mathbf{x}) \leq \bar{\alpha}_2(|\mathbf{x}|), \tag{5.15}$$

$$\tfrac{\partial V}{\partial \mathbf{x}}\mathbf{f}(\mathbf{x}) \leq -\bar{\alpha}_3(\mathbf{x}), \tag{5.16}$$

where $\bar{\alpha}_1, \bar{\alpha}_2 \in \mathscr{K}_\infty$ and $\bar{\alpha}_3(\mathbf{x})$ is a positive semidefinite function.

Define a positive and non-decreasing function $\bar{\alpha}_4$ and a class $\mathscr{K}$ function π as follows

$$\bar{\alpha}_4(r) = \sup_{|\mathbf{x}|\le r} \left| \frac{\partial V}{\partial \mathbf{x}} \right|,$$

and

$$\pi(r) = \int_0^r \frac{1}{1+\chi(s)} ds,$$

where $\chi(s)$ is a positive and non-decreasing function to be defined later.

Let $W(\mathbf{x}) = \pi(V(\mathbf{x}))$. Therefore,

$$\begin{aligned} \frac{\partial W}{\partial \mathbf{x}} \dot{\mathbf{x}} &= \frac{\partial \pi}{\partial V} \frac{\partial V}{\partial \mathbf{x}} \mathbf{f}(\mathbf{x}) + \frac{\partial \pi}{\partial V} \frac{\partial V}{\partial \mathbf{x}} \mathbf{g}(\mathbf{x}) \mathbf{u} \\ &\le \frac{-\bar{\alpha}_3(\mathbf{x})}{1+\chi(V(\mathbf{x}))} + \frac{\bar{\alpha}_4(|\mathbf{x}|)|\mathbf{g}(\mathbf{x})|\,|\mathbf{u}|}{1+\chi(V(\mathbf{x}))}. \end{aligned} \tag{5.17}$$

Define $\chi(V(\mathbf{x})) = \bar{\alpha}_4(\bar{\alpha}_1^{-1}(V(\mathbf{x})))$, which is positive and non-decreasing.

From (5.15), we have $|\mathbf{x}| \le \bar{\alpha}_1^{-1}(V(\mathbf{x}))$ and $V(\mathbf{x}) \le \bar{\alpha}_2(|\mathbf{x}|)$. As $\bar{\alpha}_4$ is a positive and non-decreasing function, we have $\bar{\alpha}_4(|\mathbf{x}|) \le \bar{\alpha}_4(\bar{\alpha}_1^{-1}(V(\mathbf{x})))$. Similarly, $\bar{\alpha}_4(\bar{\alpha}_1^{-1}(V(\mathbf{x}))) \le \bar{\alpha}_4(\bar{\alpha}_1^{-1}(\bar{\alpha}_2(|\mathbf{x}|)))$. With these inequalities, (5.17) can be written as

$$\begin{aligned} \frac{\partial W}{\partial \mathbf{x}} \dot{\mathbf{x}} &\le \frac{-\bar{\alpha}_3(\mathbf{x})}{1+\bar{\alpha}_4(\bar{\alpha}_1^{-1}(V(\mathbf{x})))} + \frac{\bar{\alpha}_4(|\mathbf{x}|)|\mathbf{g}(\mathbf{x})|\,|\mathbf{u}|}{1+\bar{\alpha}_4(\bar{\alpha}_1^{-1}(V(\mathbf{x}))} \\ &\le \frac{-\bar{\alpha}_3(\mathbf{x})}{1+\bar{\alpha}_4(\bar{\alpha}_1^{-1}(\bar{\alpha}_2(|\mathbf{x}|)))} + \frac{\bar{\alpha}_4(|\mathbf{x}|)|\mathbf{g}(\mathbf{x})|\,|\mathbf{u}|}{1+\bar{\alpha}_4(|\mathbf{x}|)} \\ &\le \frac{-\bar{\alpha}_3(\mathbf{x})}{1+\bar{\alpha}_4(\bar{\alpha}_1^{-1}(\bar{\alpha}_2(|\mathbf{x}|)))} + |\mathbf{g}(\mathbf{x})|\,|\mathbf{u}|. \end{aligned} \tag{5.18}$$

Therefore, set $\alpha_1 = \pi(\bar{\alpha}_1)$, $\alpha_2 = \pi(\bar{\alpha}_2)$, $\alpha_3 = \frac{-\bar{\alpha}_3(\mathbf{x})}{1+\bar{\alpha}_4(\bar{\alpha}_1^{-1}(\bar{\alpha}_2(|\mathbf{x}|)))}$, and $\gamma = \|\mathbf{g}(\mathbf{x})\|$, this completes the proof. ■

The constructions of $\pi(r)$ and $W(\mathbf{x})$ are motivated by the techniques in Angeli et al. (2000b).

Lemma 5.3 shows if the unforced system is Lyapunov stable, there exists a continuously differentiable function $W(\mathbf{x})$ satisfying the similar conditions (5.10) and (5.11) in Theorem 5.1. As Theorem 5.1 provides a sufficient condition for P-type learning rule convergence, comparing the results in Lemma 5.3 and conditions in Theorem 5.1, it shows that when α_1 in Lemma 5.3 happens to be a $\mathscr{K}_\infty$ function and convex, all the conditions in Theorem 5.1 are fulfilled. Therefore, we have the following corollary.

Corollary 5.1 *If α_1 in Lemma 5.3 is a $\mathscr{K}_\infty$ function and convex, then the P-type learning rule (5.3) is convergent.*

The proof of Lemma 5.3 and Corollary 5.1 provide a constructive method to check if the P-type learning rule works for a Lyapunov stable system. Next, we demonstrate the applications of Corollary 5.1 by two examples, namely, stable system with quadratic Lyapunov function and globally exponentially stable system. As the results cannot be obtained by existing analysis method in the literature, they are presented in the form of corollaries.

Corollary 5.2 *Consider system (5.1) under the P-type learning rule (5.3), Assumptions 5.1 and 5.2. If the unforced system of (5.1a) is Lyapunov stable, and admits a quadratic Lyapunov function, that is $V(\mathbf{x}) = \mathbf{x}^T P\mathbf{x}$ where P is a symmetric positive definite matrix, then the P-type learning rule (5.3) is convergent.*

Proof: As the unforced system of (5.1a) admits a quadratic Lyapunov function $V(\mathbf{x}) = \mathbf{x}^T P\mathbf{x}$, we can construct two $\mathscr{K}_\infty$ functions $\bar{\alpha}_1$ and $\bar{\alpha}_2$ such that

$$\bar{\alpha}_1(|\mathbf{x}|) \leq V(\mathbf{x}) \leq \bar{\alpha}_2(|\mathbf{x}|),$$

where $\bar{\alpha}_1(r) = c_1 r^2$, $\bar{\alpha}_2(r) = c_2 r^2$, $c_1 = \lambda_{min}(P)$, and $c_2 = \lambda_{max}(P)$.

$\bar{\alpha}_4$ can be calculated as follows,

$$\bar{\alpha}_4(r) = \sup_{|\mathbf{x}|\leq r} \left| \frac{\partial V(\mathbf{x})}{\partial \mathbf{x}} \right| = 2|P|r = c_4 r.$$

Following the similar procedures in the proof of Lemma 5.3, $W(\mathbf{x})$ can be constructed by $\pi(V(\mathbf{x}))$. Therefore, $\alpha_1 = \pi(\bar{\alpha}_1)$. Utilizing Lemma 5.3 and Corollary 5.1, it suffices to show α_1 is a $\mathscr{K}_\infty$ function and convex so as to conclude Corollary 5.2. Calculating α_1, we have

$$\begin{aligned}\alpha_1(r) &= \int_0^{\bar{\alpha}(r)} \frac{1}{1+\bar{\alpha}_4(\bar{\alpha}_1^{-1}(s))} ds \\ &= \int_0^{c_1 r^2} \frac{1}{1+\frac{c_4}{\sqrt{c_1}}\sqrt{s}} ds. \end{aligned} \tag{5.19}$$

From (5.19), we can conclude that $\alpha_1(r)$ is nonnegative, continuous, monotonically increasing, and approaches to infinity as $r \to \infty$. Thus, $\alpha_1 \in \mathscr{K}_\infty$.

Differentiating α_1 twice against r yields,

$$\frac{d\alpha_1^2}{d^2r} = \frac{2c_1}{(1+c_4 r)^2} > 0.$$

Therefore, α_1 is a convex function. This completes the proof. ■

As a specific example, consider the dynamics below

$$\dot{x} = -x - x^3 + u, \tag{5.20a}$$

$$y = x + u, \tag{5.20b}$$

The system (5.20) contains a local Lipschitz term $-x^3$, all existing ILC theory does not apply to this system. It is straightforward to verify that the unforced system of (5.20a) admits the Lyapunov function $V(x) = x^2$. Thus, by Corollary 5.2, P-type learning rule is applicable to system (5.20).

Corollary 5.3 *Consider system (5.1) under the P-type learning rule (5.3), Assumptions 5.1 and 5.2. If the unforced system of (5.1a) is globally exponentially stable, then the P-type learning rule (5.3) is convergent.*

Proof: As the unforced system of (5.1a) is globally exponentially stable, by Converse Theorem (Khalil, 2002), there exists a Lyapunov function $V(\mathbf{x})$ such that

$$c_1|\mathbf{x}|^2 \leq V(\mathbf{x}) \leq c_2|\mathbf{x}|^2,$$

$$\frac{\partial V}{\partial \mathbf{x}}\mathbf{f}(\mathbf{x}) \leq -c_3|\mathbf{x}|^2,$$

$$\left|\frac{\partial V}{\partial \mathbf{x}}\right| \leq c_4|\mathbf{x}|,$$

where c_1, c_2, c_3, c_4 are positive constants. Therefore, we can draw the conclusion by using similar arguments in the proof of Corollary 5.2. ■

It is easy to show that the unforced system of (5.20a) is globally exponentially stable, Corollary 5.3 says P-type learning rule is applicable to system (5.20), which is the same as predicted by Corollary 5.2. Corollary 5.3 can be regarded as a special case of Corollary 5.2.

5.3.3 Systems with Stable Local Lipschitz Terms but Unstable Global Lipschitz Factors

All the results presented so far in this section assume Lyapunov stable unforced systems. However, it is widely observed that ILC works for unstable systems as well, especially the unstable system that are globally Lipschitz continuous. This motivates us to further explore what kind of unstable system is applicable for ILC. The following Lyapunov criterion renders the existing ILC for global Lipschitz system as a special case.

Theorem 5.2 *Consider system (5.1) under the P-type learning rule (5.3), Assumptions 5.1 and 5.2. If there exist $\alpha_1, \alpha_2 \in \mathcal{K}_\infty$, a continuously differentiable function $W(\mathbf{x})$,* *and positive constants γ_1, γ_2, γ_3, such that*

$$\alpha_1(|\mathbf{x}|) \leq W(\mathbf{x}) \leq \alpha_2(|\mathbf{x}|) \tag{5.21}$$

$$\tfrac{\partial W}{\partial \mathbf{x}}(\mathbf{f}(\mathbf{x}) + \mathbf{g}(\mathbf{x})\mathbf{u}) \leq \gamma_1|\mathbf{x}| + \gamma_2|\mathbf{u}| + \gamma_3. \tag{5.22}$$

Furthermore, if α_1 is convex, then the P-type learning rule (5.3) is convergent.

Proof: From (5.22) we have

$$W(\mathbf{x}(t)) \leq \gamma_1 \int_0^t |\mathbf{x}(\tau)|\,d\tau + \gamma_2 \int_0^t |\mathbf{u}(\tau)|\,d\tau + \gamma_3 t + W(\mathbf{x}(0)). \tag{5.23}$$

Noticing equation (5.21) and Lemma 5.1, we have following relation,

$$|\mathbf{x}(t)| \leq \gamma_1 k \int_0^t |\mathbf{x}(\tau)|\,d\tau + \gamma_2 k \int_0^t |\mathbf{u}(\tau)|\,d\tau + \gamma_3 kt + b, \tag{5.24}$$

for $t \in [0, T]$, where k depends on α_1, and b are positive constants that depend on α_1 and $W(\mathbf{x}(0))$.

Taking λ-norm on both sides of (5.24) yields

$$\|\mathbf{x}\|_\lambda \quad \leq \quad \frac{\gamma_1 k}{\lambda}\|\mathbf{x}\|_\lambda + \frac{\gamma_2 k}{\lambda}\|\mathbf{u}\|_\lambda + \gamma_3 kT + b.$$

Solving $\|\mathbf{x}\|_\lambda$ in the above equation yields

$$\|\mathbf{x}\|_\lambda \quad \leq \quad \left(1 - \frac{\gamma_1 k}{\lambda}\right)^{-1} \left(\frac{\gamma_2 k}{\lambda}\|\mathbf{u}\|_\lambda + \gamma_3 kT + b\right). \tag{5.25}$$

The relation between $\mathbf{u}_{i+1}$ and $\mathbf{u}_i$ has been derived in (5.7). Similarly, taking λ-norm on both sides of (5.7) yields

$$\|\mathbf{u}_{i+1}\|_\lambda \leq \rho\|\mathbf{u}_i\|_\lambda + c|\Gamma|\|\mathbf{x}_i\|_\lambda + d|\Gamma|\|\mathbf{u}_d\| + c|\Gamma|\|\mathbf{x}_d\|, \tag{5.26}$$

where c and d are positive constants defined in the proof of Lemma 5.2.

Substituting (5.25) to (5.26), we can obtain

$$\|\mathbf{u}_{i+1}\|_\lambda \leq \left(\rho + \left(1-\tfrac{\gamma_1 k}{\lambda}\right)^{-1} \tfrac{c\gamma_2 k|\Gamma|}{\lambda}\right)\|\mathbf{u}_i\|_\lambda$$
$$\underbrace{+c|\Gamma|\left(1-\frac{\gamma_1 k}{\lambda}\right)^{-1}(\gamma_3 kT+b)+d|\Gamma|\|\mathbf{u}_d\|+c|\Gamma|\|\mathbf{x}_d\|}_{\text{iteration-invariant}}. \tag{5.27}$$

Choose a sufficiently large λ such that

$$\rho + \left(1-\frac{\gamma_1 k}{\lambda}\right)^{-1}\frac{c\gamma_2 k|\Gamma|}{\lambda} = \bar{\rho} < 1.$$

Hence, $\mathbf{u}_i$ and $\mathbf{x}_i$ are bounded for all iterations. Follow the similar idea in the proof of Lemma 5.2, we can conclude Theorem 5.2. ■

From Theorem 5.2 we can conclude that P-type learning rule is applicable to a system whose local Lipschitz terms are stable and global Lipschitz terms can be unstable. Next, we show that Theorem 5.2 includes global Lipschitz system as a special. Consider system (5.1), and assume that $\mathbf{f}(\mathbf{x})$ satisfies the global Lipschitz condition with Lipschitz constant L. Let $\pi(r)=\int_0^r \frac{1}{1+\sqrt{s}}ds$, $V(\mathbf{x})=\mathbf{x}^T\mathbf{x}$, and $W(\mathbf{x})=\pi(V(\mathbf{x}))$. Thus, $W(\mathbf{x})$ can be written as

$$W(\mathbf{x}) = \int_0^{\mathbf{x}^T\mathbf{x}} \frac{1}{1+\sqrt{s}}ds.$$

In this case $\bar{\alpha}_1(r)=r^2$. Note that we have already shown that $\pi(\bar{\alpha}_1)$ is a $\mathscr{K}_\infty$ function and convex in the proof to Corollary 5.2.

Differentiating $W(\mathbf{x})$ yields,

$$\begin{aligned}\frac{\partial W}{\partial \mathbf{x}}(\mathbf{f}(\mathbf{x})+\mathbf{g}(\mathbf{x})\mathbf{u}) &= \frac{2\mathbf{x}^T\mathbf{f}(\mathbf{x})+2\mathbf{x}^T\mathbf{g}(\mathbf{x})\mathbf{u}}{1+|\mathbf{x}|}\\ &\leq 2L|\mathbf{x}|+2\|\mathbf{g}(\mathbf{x})\||\mathbf{u}|+2|\mathbf{f}(0)|.\end{aligned}$$

Set $\gamma_1=2L$, $\gamma_2=2\|\mathbf{g}(\mathbf{x})\|$, and $\gamma_3=2|\mathbf{f}(0)|$. Hence, all global Lipschitz systems satisfy the condition in Theorem 5.2.

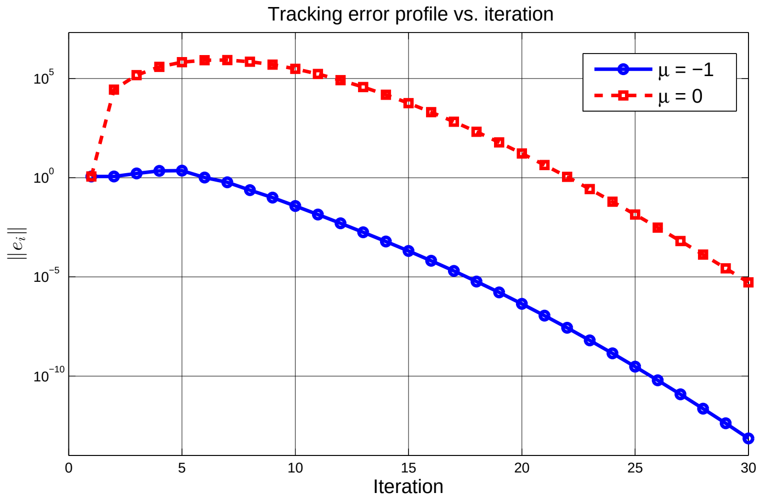

Figure 5.1: Tracking error profiles vs. iteration number for $\mu = -1$ and $\mu = 0$.

Consider the dynamics below

$$\dot{x} = 2x + \mu x^3 + u, \tag{5.28a}$$

$$y = x + u, \tag{5.28b}$$

The unforced system of (5.28a) has an unstable origin and contains a local Lipschitz term μx^3. It is easy to show that P-type learning rule is applicable to system (5.28) by Theorem 5.2 when $\mu < 0$. In contrast, all existing ILC theories cannot claim such a conclusion unless $\mu = 0$. To verify the theoretical prediction, the following P-type controller is applied to system (5.28)

$$u_{i+1} = u_i + e_i,$$

where $e_i = y_d - y_i$, and $y_d(t) = 0.1t + \sin(t)$ for $t \in [0, 5]$. The initial state is set to $x_i(0) = 0$ for all iterations, and the control input in the first iteration $u_i(t) = 0$. It can be seen from Figure. 5.1 that the P-type learning rule converges for both systems, i.e., $\mu = -1$ and $\mu = 0$. A closer examination shows when $\mu = 0$ the tracking error initially increases up to the magnitude of 10^5, and then exponentially converges to zero. In contrast, the transient performance for the system when $\mu = -1$ is much smoother. This is because the local Lipschitz term $-x^3$ is stable and it prevents the system state from growing to a large magnitude. This observation also suggests that it may help improve the transient performance by stabilizing a system before applying the ILC rule.

5.4 Convergence Properties in Presence of Bounding Conditions

5.4.1 Systems with Bounded Drift Term

In addition to Assumption 5.1, we further assume that the drift term $\mathbf{f}(\mathbf{x})$ in system (5.1) is bounded for all $\mathbf{x} \in \mathbb{R}^n$. Then, we have the following theorem.

Theorem 5.3 *Consider system (5.1) under the P-type learning rule (5.3), Assumptions 5.1 and 5.2. If* $\|\mathbf{f}(\mathbf{x})\| = \sup_{\mathbf{x}\in\mathbb{R}^n} |\mathbf{f}(\mathbf{x})| \leq b_f$ *and* b_f *is a positive constant, then the P-type learning rule (5.3) is convergent.*

Proof: From the system dynamics (5.1a), the system state can be written as

$$\mathbf{x}(t) = \mathbf{x}(0) + \int_0^t (\mathbf{f}(\mathbf{x}(\tau)) + \mathbf{g}(\mathbf{x}(\tau))\mathbf{u}(\tau))\, d\tau. \tag{5.29}$$

Taking norm on both sides of (5.29) and noticing the boundedness assumption on $\mathbf{f}(\mathbf{x})$, a bounding function on $|\mathbf{x}(t)|$ can be obtained

$$|\mathbf{x}(t)| \leq \|\mathbf{g}(\mathbf{x})\| \int_0^t |\mathbf{u}(\tau)|\, d\tau + b_f t + |\mathbf{x}(0)|. \tag{5.30}$$

Note that the bounding function on $|\mathbf{x}(t)|$ in (5.30) has the same form as in (5.5). Thus, we can conclude that the P-type learning rule (5.3) is convergent based on Lemma 5.2.

■

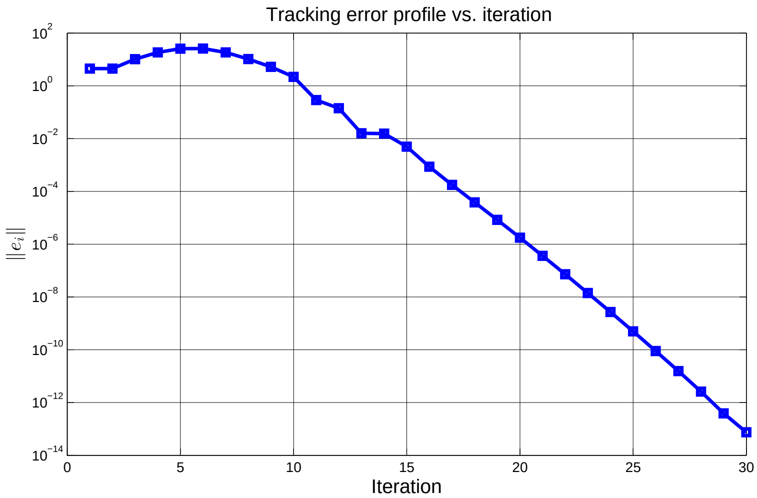

Figure 5.2: Tracking error profiles vs. iteration number for system with bounded local Lipschitz term.

To verify the result in Theorem 5.3, consider the following system

$$\dot{x} = \sin(x^2) + u,$$

$$y = x + u.$$

Note that the local Lipschitz term $\sin(x^2)$ is bounded for all $x \in \mathbb{R}$. Let the desired control input be $u_d(t) = 0.1t + 1.5\sin(2t)$, the initial state $x_i(0) = 0$, and $u_1(t) = 0$ for $t \in [0, 5]$. Applying the P-type learning rule

$$u_{i+1} = u_i + e_i,$$

it can be seen from Figure. 5.2 that the tracking error increases slightly and then exponentially converges to zero.

5.4.2 Systems with Bounded Control Input

In almost all practical systems, the inputs always have finite power due to physical limitations. In this subsection, we apply control saturation to a class of uniformly bounded energy bounded state (UBEBS) systems (Angeli et al., 2000a), and explore the applicability of P-type ILC rule to UBEBS systems.

Definition 5.1 *The system (5.1) is UBEBS if there exist class $\mathscr{K}_\infty$ functions α_1, α_2, and some positive constant a such that*

$$|\mathbf{x}(t)| \le \int_0^t \alpha_1(|\mathbf{u}(\tau)|)\,d\tau + \alpha_2(|\mathbf{x}(0)|) + a,$$

for any $t \in [0, T]$.

The definition of UBEBS shares the similar but more general definitions than ISS (Jiang and Wang, 2001; Khalil, 2002) and iISS (Angeli et al., 2000b; Sontag, 2006). The system (5.1) is said to be ISS if the state satisfies

$$|\mathbf{x}(t)| \le \beta(|\mathbf{x}(0)|, t) + \gamma\|\mathbf{u}\|,$$

where $\beta(|\mathbf{x}(0)|, t)$ is a class $\mathscr{K}\mathscr{L}$ function[2], and γ is a positive constant.

The system (5.1) is said to be iISS if the state satisfies

$$\alpha_1(|\mathbf{x}(t)|) \le \beta(|\mathbf{x}(0)|, t) + \int_0^t \alpha_2(|\mathbf{u}(\tau)|)\,d\tau,$$

where α_1 and α_2 are class $\mathscr{K}$ functions.

Based on the definitions of UBEBS, ISS and iISS, we can conclude that UBEBS systems include ISS and iISS systems as special cases. Both ISS and iISS require that the system has a stable origin. However, UBEBS system does not require a stable origin. Therefore, the results derived in this subsection can be applied to a large class of systems when control saturation is imposed.

[2] $\beta(s,t)$ is said to belong to class $\mathscr{K}\mathscr{L}$ if for each fixed t, the mapping $\beta(s,t)$ belongs to class $\mathscr{K}$ with respect to s, and for each fixed s, the mapping $\beta(s,t)$ is decreasing with respect to t and $\beta(s,t) \to 0$ as $t \to \infty$.

The P-type learning rule with control saturation is proposed as follows

$$\mathbf{v}_{i+1} = \mathbf{u}_i + \Gamma \mathbf{e}_i, \tag{5.32a}$$

$$\mathbf{u}_{i+1} = \mathrm{sat}(\mathbf{v}_{i+1}, \mathbf{u}^*) \tag{5.32b}$$

where Γ is chosen such that (5.4) is satisfied, the saturation function $\mathrm{sat}(\cdot, \mathbf{u}^*)$ is defined component-wise, the saturation bound

$$\mathbf{u}^* = \begin{bmatrix} \underline{u_1}, & \overline{u_1} \\ \underline{u_2}, & \overline{u_2} \\ \vdots & \vdots \\ \underline{u_p}, & \overline{u_p} \end{bmatrix},$$

and $\underline{u_j}, \overline{u_j}$ denote the lower and upper limits of the jth input channel and $1 \le j \le p$.

As the desired trajectory $\mathbf{y}_d$ in (5.2) should be realizable, $\mathbf{u}_d$ must not exceed the saturation bound, that is $\mathbf{u}_d = \mathrm{sat}(\mathbf{u}_d, \mathbf{u}^*)$.

Applying the control saturation (5.32b), the input never exceeds the saturation bound. Therefore, the upper bound on state $|\mathbf{x}(t)|$ can be obtained from Definition 5.1. Denote the upper bound of $|\mathbf{x}(t)|$ by $b_x = b_x(\mathbf{x}(0), \mathbf{u}^*, T)$. As such, $\mathbf{f}(\mathbf{x})$ can be regarded as a global Lipschitz function in $\mathbf{x}$ with Lipschitz constant L, and

$$L = \sup_{\|\mathbf{z}_1\| < b_x, \|\mathbf{z}_2\| < b_x} \phi(\mathbf{z}).$$

Theorem 5.4 *Consider system (5.1) under the P-type learning rule (5.32), Assumptions 5.1 and 5.2. If the system (5.1) is UBEBS, then the P-type learning rule (5.32) with control saturation is convergent.*

Proof: As $\mathbf{g}(\mathbf{x})$ is a continuously differentiable function and $\|\mathbf{x}\| = b_x$, the following inequality holds

$$|\mathbf{g}(\mathbf{x}_d) - \mathbf{g}(\mathbf{x}_i)| \le L_g |\mathbf{x}_d - \mathbf{x}_i|, \tag{5.33}$$

where L_g is a constant.

Let $\delta\mathbf{x}_i = \mathbf{x}_d - \mathbf{x}_i$ and $\delta\mathbf{u}_i = \mathbf{u}_d - \mathbf{u}_i$. Note that $\delta\mathbf{x}_i(0) = 0$ by Assumption 5.2, thus

$$\begin{aligned}\delta\mathbf{x}_i(t) &= \int_0^t (\mathbf{f}(\mathbf{x}_d) - \mathbf{f}(\mathbf{x}_i) + \mathbf{g}(\mathbf{x}_d)\mathbf{u}_d - \mathbf{g}(\mathbf{x}_i)\mathbf{u}_i)\, d\tau \\ &= \int_0^t (\mathbf{f}(\mathbf{x}_d) - \mathbf{f}(\mathbf{x}_i) + \mathbf{g}(\mathbf{x}_d)\mathbf{u}_d - \mathbf{g}(\mathbf{x}_i)\mathbf{u}_d + \mathbf{g}(\mathbf{x}_i)\mathbf{u}_d - \mathbf{g}(\mathbf{x}_i)\mathbf{u}_i)\, d\tau. \quad (5.34)\end{aligned}$$

Taking norm on both sides of (5.34) yields

$$|\delta\mathbf{x}_i(t)| \le \int_0^t ((L + L_g\|\mathbf{u}_d\|)|\delta\mathbf{x}_i| + \|\mathbf{g}(\mathbf{x}_i)\||\delta\mathbf{u}_i|)\, d\tau. \quad (5.35)$$

Applying Gronwall-Bellman's Lemma to (5.35) yields

$$|\delta\mathbf{x}_i(t)| \le \int_0^t e^{(L+L_g\|\mathbf{u}_d\|)(t-\tau)}\|\mathbf{g}(\mathbf{x}_i)\||\delta\mathbf{u}_i(\tau)|\, d\tau. \quad (5.36)$$

Taking λ-norm on (5.36) we obtain

$$\|\delta\mathbf{x}_i\|_\lambda \le \frac{b_g}{\lambda - l}\|\delta\mathbf{u}_i\|_\lambda, \quad (5.37)$$

where $l = L + L_g\|\mathbf{u}_d\|$ and $b_g = \|\mathbf{g}(\mathbf{x})\|$.

Next, calculate the relation between $\delta\mathbf{u}_{i+1}$ and $\delta\mathbf{u}_i$.

$$\begin{aligned}|\delta\mathbf{u}_{i+1}| &= |\mathbf{u}_d - \mathrm{sat}(\mathbf{v}_{i+1}, \mathbf{u}^*)| \\ &\le |\mathbf{u}_d - \mathbf{v}_{i+1}| \\ &= |\mathbf{u}_d - \mathbf{u}_i - \Gamma\mathbf{e}_i| \\ &= \left|\left(I_m - \Gamma\frac{\partial\mathbf{h}}{\partial\mathbf{u}}\right)\delta\mathbf{u}_i + \frac{\partial\mathbf{h}}{\partial\mathbf{x}}\delta\mathbf{x}_i\right| \\ &\le \rho|\delta\mathbf{u}_i| + c|\delta\mathbf{x}_i|, \quad (5.38)\end{aligned}$$

where c is defined in the proof to Lemma 5.2.

Note that (5.7) is utilized in deriving (5.38). Taking λ-norm on (5.38) yields

$$\|\delta\mathbf{u}_{i+1}\|_\lambda \le \rho\|\delta\mathbf{u}_i\|_\lambda + c\|\delta\mathbf{x}_i\|_\lambda. \quad (5.39)$$

Substituting (5.37) to (5.39), we obtain

$$\|\delta\mathbf{u}_{i+1}\|_\lambda \leq \left(\rho + \frac{cb_g}{\lambda - l}\right)\|\delta\mathbf{u}_i\|_\lambda. \tag{5.40}$$

Choose λ sufficiently large such that (5.40) is a contraction-mapping. Therefore, $\|\delta\mathbf{u}_i\|$ converges to zero in the iteration domain. Consequently, $\|\delta\mathbf{x}_i\|$ converges to zero as well. This completes the proof. ■

To demonstrate the application of Theorem 5.4, we consider the switched reluctance motor (SRM) control problem, which is one of the benchmark problems in ILC literature (Sahoo et al., 2004). In a SRM, the rotor is made of steel laminations without conductors or permanent magnets, and only the stator presents windings. Due to the simple mechanical structure, SRM is cheaper to manufacture compared with other ac and dc motors. However, the presence of nonlinearity makes SRM a challenging control problem.

According to Spong et al. (1987), an m-phase SRM can be modeled by the following dynamics,

$$\begin{aligned}
\dot{\theta}_e &= \omega_e, \\
\dot{\omega}_e &= J^{-1}\left(\sum_{j=1}^{m} T_j(\theta_e, \eta_j) - f\omega_e\right), \\
T_j &= \frac{\psi_s}{h_j^2(\theta_e)}\frac{d\,h_j(\theta_e)}{d\,\theta_e}\left(1 - [1 + \eta_j h_j(\theta_e)]e^{-\eta_j h_j(\theta_e)}\right), \\
\dot{\eta}_j &= \left(\frac{\partial\psi_j}{\partial\eta_j}\right)^{-1}\left(-R\eta_j + u_j - \frac{\partial\psi_j}{\partial\theta_e}\omega_e\right), \\
\psi_j &= \psi_s\left(1 - e^{\eta_j h_j(\theta_e)}\right), \\
h_j(\theta_e) &\approx h_0 + b_1\sin(N_r\theta_e - 2\pi(j-1)/m),
\end{aligned}$$

where $j = 1, \ldots, m$ denotes the phase, $m = 3$, θ_e is the electrical angular position of rotor, ω_e is the angular speed, $N_r = 4$ is the number of rotor poles, η_j is the stator current of phase j, u_j is the voltage applied on phase j, ψ_j is the flux-linkage of phase

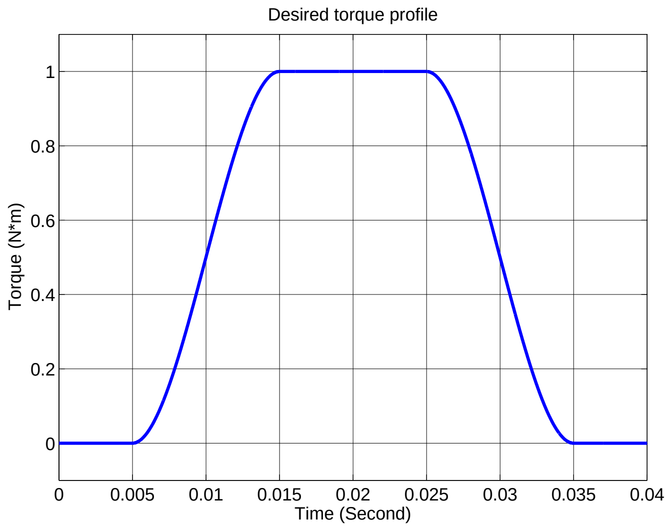

Figure 5.3: Desired torque profile.

j, $\psi_s = 1.2$Wb is the flux constant, $J = 0.0016\text{kgm}^2$ is the inertial of rotor, $R = 4\Omega$ is resistance of phase j, $f = 0.2$ is the friction coefficient, $h_0 = 0.0545$ and $b_1 = 0.0454$ are two constants.

To illustrate the P-type learning rule, consider only the current to torque loop. Let the desired torque profile be $T_d(t) \in \mathscr{C}[0, 0.04]$ as follows,

$$T_d(t) = \begin{cases} 0 & 0 \le t < 0.05 \\ \frac{3}{\tau^2}(t-0.005)^2 - \frac{2}{\tau^3}(t-0.005)^3 & 0.005 \le t < 0.015 \\ 1 & 0.015 \le t < 0.025 \\ 1 - \frac{3}{\tau^2}(t-0.025)^2 + \frac{2}{\tau^3}(t-0.025)^3 & 0.025 \le t < 0.035 \\ 0 & 0.035 \le t \le 0.04 \end{cases}$$

where $\tau = 0.01$. Figure. 5.3 shows the desired torque profile.

Due to the symmetric structure, apply learning rule to the first phase only, and the controller is chosen as

$$\eta_{1,i+1} = \text{sat}(\eta_{1,i} + \gamma e_i, \eta^*),$$

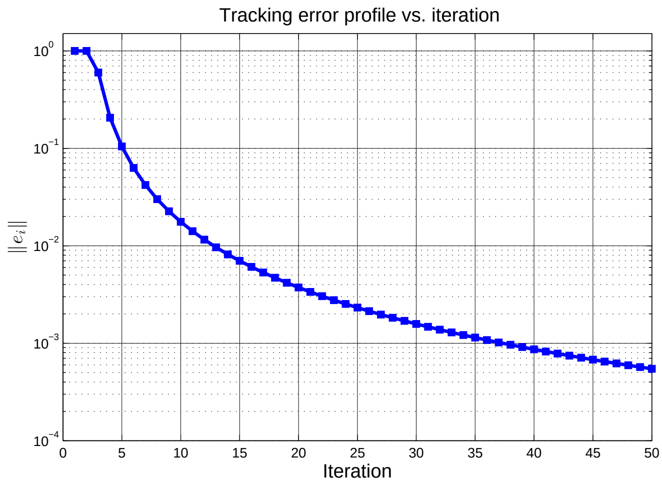

Figure 5.4: Tracking error profiles vs. iteration number under control saturation.

where i denotes the iteration number, learning gain $\gamma = 2$, tracking error $e_i = T_d - T_{1,i}$, and saturation bound $\eta^* = [-20, 20]$. The application of control saturation is reasonable as in any electrical drives a current limiter is always incorporated to prevent the circuit from current overflow or overheating.

In the simulation study, the initial conditions are chosen as $\theta_e = 0$, $\omega_e = 0$, and $\eta_{1,0} = 0$. Figure. 5.4 shows the tracking error profiles in the iteration domain. After 37 iterations of learning, the tracking error is reduced to 10^{-3}.

5.5 Conclusion

This chapter explores the applicability of P-type ILC rule to local Lipschitz systems. In the current ILC literature, contraction-mapping based ILC is only applicable to global Lipschitz systems. In contrast, a number of sufficient conditions for local Lipschitz systems in the form of Lyapunov criteria are developed in the current work. By using the similar proof idea, it is further shown that P-type ILC can be applied to the systems whose local Lipschitz terms are bounded, and UBEBS systems under control saturation.

The new development complements to the existing literature. However, there are many issues remaining unsolved. In particular, the applicability of D-type ILC rule to local Lipschitz systems is still unknown, and is worth more investigation.

Chapter 6

Synchronization for Nonlinear Multi-agent Systems by Adaptive Iterative Learning Control

6.1 Background

As we discussed before, there are two main frameworks for iterative learning control, namely contraction-mapping (CM) and composite energy function (CEF) based approaches. CM based ILC is usually preferred due to its simplicity in implementation, little system information requirement, and applicability to non-affine in input systems. However, in the current literature, CM based ILC is only applicable to the global Lipschitz systems. In Chapter 5 CM based ILC is extended to several classes of local Lipschitz systems by combining Lyapunov and CM analysis methods. Therefore, the algorithms developed in Chapters 2, 3, and 4 can be applied to a large scale of systems. In contrast, CEF based method uses full state information to design estimation rule and

ILC controllers. This method can be applied if the system dynamics have the linear in parameter form and full state information is available for feedback or nonlinear compensation. As the current state information is utilized in the controller, the transient performance is usually better than the CM based method.

Recently Yang and Xu (2012) demonstrate promising results of the CEF method in consensus tracking problem for the first-order parametric agents. Li and Li (2013) apply the similar approach to the second-order agents. The focus of this chapter is to design ILC schemes for agents that have more general nonlinear dynamics. The CEF framework is used to analyze the performance of synchronization algorithms. Comparing to the existing literature, the non-parametric uncertainties in Li and Li (2013) are assumed to be bounded and suppressed by sliding mode control. In this chapter, it is assumed that the non-parametric uncertainties satisfy the local Lipschitz like condition and could be unbounded. They are carefully handled by some robust terms. In Yang and Xu (2012) and Li and Li (2013), the problem formulations assume constant and known input gains. Whereas, the time-varying input gains with uncertainties are considered in this chapter. Moreover, not restricted to the first-order or second-order systems, the developed methods can be applied to any high-order systems.

In a traditional ILC setting (tracking problem), the desired trajectory is used to design updating laws. However, in the multi-agent coordination problem, it is very common that the desired trajectory is only available to a subset of followers. Hence, only the local measurement (the extended error) can be utilized in the controller design. Although there is a clear relationship between the extended error and the consensus tracking error, it is not straightforward to apply the CEF based design framework. In this chapter, the desired trajectory is treated as unknown time-varying (iteration-invariant) parametric uncertainty so that each follower tries to learn. This desired trajectory is combined with other parametric uncertainties that ILC can handle easily. On the other hand, the nonlinear uncertainties that are not global Lipschitz continuous will be handled by robust control. The proposed algorithm enables all the followers to learn the parametric uncertainties and deal with lumped uncertainties based on local information from their neighborhoods. The convergence analysis using an appropriate CEF shows that the proposed ILC algorithms can achieve consensus tracking in the presence of parametric and lumped uncertainties under some appropriate convergence conditions.

The rest of this chapter is organized as follows. Section 6.2 provides one preliminary result in graph theory and the problem formulation. Next, ILC control laws are designed for the first-order multi-agent systems in Section 6.3. Section 6.4 extends the obtained results to the more general settings: high-order systems. Illustrative examples are presented in Section 6.5 to demonstrate efficacy of the proposed algorithms, followed by conclusion in Section 6.6.

6.2 Preliminaries and Problem Description

6.2.1 Preliminaries

Graph theory is an instrumental tool to characterize the communication topology in the multi-agent setting. Unlike the previous chapters, an undirected graph $\mathscr{G}(\mathscr{V},\mathscr{E})$ is adopted to model the communication among followers. That is the information flow among followers is symmetric. The following lemma is a well-known result in multi-agent coordination, and it is an important result to construct an appropriate CEF.

Lemma 6.1 *(Hong et al., 2006) If the undirected graph $\mathscr{G}$ is connected, and D is any nonnegative diagonal matrix with at least one of the diagonal entries being positive,* *then $H = L + D$ is symmetric positive definite, where L is the Laplacian matrix of $\mathscr{G}$.*

6.2.2 Problem description for first-order systems

The problem formulation starts from simple first-order systems. This simplifies the design of ILC updating law. Section 6.4 shows how to generalize the obtained results to high-order systems.

Consider a group of N heterogeneous agents. At the ith iteration, the dynamics of the jth agent, $(j = 1,2,\ldots,N)$, take the following form,

$$\dot{x}_{i,j}(t) = \theta_j^0(t)\xi_j^0(t,x_{i,j}) + \eta_j(t,x_{i,j}) + b_j(t)u_{i,j}(t), \tag{6.1}$$

where $x_{i,j} \in \mathbb{R}^1$ is the system state, $u_{i,j} \in \mathbb{R}^1$ is the input, $\theta_j^0 : [0,T] \to \mathbb{R}^{1\times n_j}$ is an unknown time-varying function, $\xi_j^0 : [0,T] \times \mathbb{R}^1 \to \mathbb{R}^{n_j}$ is a state-dependent known function, $\eta_j : [0,T] \times \mathbb{R}^1 \to \mathbb{R}^1$ represents the unknown lumped system uncertainties or disturbances. The parameter $b_j \in \mathbb{R}^1$ is the time-varying input gain, and it is either positive or negative. Without loss of generality, it is assumed that $0 < \underline{b}_j \leq b_j \leq \overline{b}_j$. In addition, $\theta_j^0(\cdot)$, $\xi_j^0(\cdot,\cdot)$, $\eta_j(\cdot,\cdot)$, and $b_j(\cdot)$ are continuous functions with respect to their arguments. For simplicity of the presentation, for a time-varying function $f(t)$, the argument t is dropped when no confusion arises.

The desired trajectory (virtual leader) satisfies $x_d \in \mathscr{C}^1[0,T]$, hence, $\dot{x}_d$ is continuous. Meanwhile, the virtual leader's trajectory $x_d(t)$, for $t \in (0,T]$ is only accessible to a small portion of the followers. Furthermore, we assume that the upper bound of the desired trajectory x_d and its initial condition x_0 are known to each follower. The supremum norm of x_d is denoted as b_d.

Remark 6.1 *Under certain scenarios, the initial states of the agents are easily obtainable compared with the states during the task execution. For example, a group of* *unmanned aerial vehicles (UAVs) take off simultaneously from an air base and perform a formation task for surveillance. Before the UAVs take off, the relative geometric positions among them can be prescribed by human operators. Whereas, during the task execution, the leader's position and velocity are not available to some of the followers due to communication or sensor limitations. Therefore, it is emphasized here that* $x_d(t)$, *for* $t \in (0,T]$ *is only accessible to a small portion of the followers.*

Remark 6.2 *For many practical systems, there are limitations due to physical constraints. For instance, the range space of an industry manipulator, the thrust on a fixed-wing aircraft, and the speed of a vehicle are all limited. As the controller requires the upper bound of* x_d*, which is the global information, we shall minimize the usage of the global information. Depending on the specific control task, the system's operation range may be used to estimate* b_d*.*

Denote the tracking error for the jth agent at the ith iteration as

$$e_{i,j} = x_d - x_{i,j}. \tag{6.2}$$

A follower can only measure or observe the state information within its neighborhood (local information). This local information is called the extended tracking error and it is defined as follows,

$$\varepsilon_{i,j} \triangleq \sum_{k\in\mathcal{N}_j} a_{j,k}(x_{i,k} - x_{i,j}) + d_j(x_d - x_{i,j}), \tag{6.3}$$

where $d_j = 1$ if the jth agent knows the desired trajectory, and $d_j = 0$ otherwise.

The control objective is to design a set of distributed controllers such that all the followers can perfectly track the desired trajectory in the presence of parametric and lumped uncertainties, i.e., $\lim_{i\to\infty} e_{i,j} = 0, \text{for } j = 1,2,\ldots,N.$

Noting the fact that $L\mathbf{1} = 0$ and using (6.2), the extended tracking error (6.3) can be written in the stack vector form,

$$\begin{aligned} \varepsilon_i &= -L\mathbf{x}_i + D\mathbf{e}_i = L(x_d\mathbf{1} - \mathbf{x}_i) + D\mathbf{e}_i \\ &= (L+D)\mathbf{e}_i = H\mathbf{e}_i, \end{aligned} \tag{6.4}$$

where $D = \mathrm{diag}(d_1, \ldots, d_N)$, $\mathbf{x}_i$, $\mathbf{e}_i$, and ε_i are the column stack vectors of $x_{i,j}$, $e_{i,j}$, and $\varepsilon_{i,j}$. If Assumption 6.2 (introduced later) holds, Lemma 6.1 shows that H is symmetric positive definite. This important feature will be used to construct an appropriate CEF.

Remark 6.3 *The relationship (6.4) plays a crucial role to ensure the consensus tracking with local information. If ε_i converges, the relationship (6.4) shows that the tracking error $\mathbf{e}_i$ will also converge as H is positive definite. This also supports the key idea of multi-agent systems: by sharing information among the communication networks, each agent does not have to equip a lot of sensors to measure needed signals.*

To simplify the discussion, the following assumptions are imposed.

Assumption 6.1 *For any $j = 1, 2, \ldots, N$, the nonlinear function $\eta_j(t,x)$ satisfies the following condition:*

$$|\eta_j(t,z_1) - \eta_j(t,z_2)| \leq \phi_j(\mathbf{z})|z_1 - z_2|, t \in [0,T], \tag{6.5}$$

where $\mathbf{z} = [z_1, z_2]^T$, and $\phi_j : \mathbb{R}^2 \to \mathbb{R}_{\geq 0}$ is a known continuous function and it is radially unbounded[1].

Remark 6.4 *Assumption 6.1 is much weaker than the global Lipschitz condition as $\phi_j(\cdot)$ might go to infinity when the state of the jth agent is unbounded. Thus the CM based ILC design framework cannot be applied in general (see Chapter 5). Although*

[1] A function $V : \mathbb{R}^n \to \mathbb{R}^1_{\geq 0}$ is radially unbounded if $|\mathbf{x}| \to \infty \Rightarrow V(\mathbf{x}) \to \infty$ (Khalil, 2002, Chapter 4).

this condition is quite weak and many nonlinear functions satisfy this condition, it is slightly stronger than local Lipschitz condition as the inequality (6.5) holds for all $\mathbf{z} \in \mathbb{R}^2$.

Assumption 6.2 *The communication graph* $\mathscr{G}$ *is undirected and connected, and at least one of the followers can access to the leader's trajectory.*

Remark 6.5 *Assumption 6.2 assumes that the leader is globally reachable from all followers. This assumption is a necessary requirement for a leader-follower consensus tracking problem. If there is an isolated agent, it is not possible that the agent can track the leader's trajectory as there is no information to correct its control action.*

Assumption 6.3 *The initial state of all followers are reset to be at* x_0 *at every iteration, i.e.,* $x_{i,j}(0) = x_d(0)$.

Remark 6.6 *Assumption 6.3 is the well-known identical initialization condition (i.i.c.), which is one of the fundamental problems in the ILC literature. It has been used in many multi-agent coordination problems, for example the formation problems considered in Ahn and Chen (2009) and Liu and Jia (2012). Without i.i.c., no matter how ILC repeats, perfect tracking can never be achieved. Many modifications have been dedicated to the relaxation of i.i.c., but they require either extra system information, or additional control mechanisms. For example the initial state learning in Yang et al. (2012). This fundamental issue has been exploited over the past 30 years without much progress. In fact, i.i.c. problem is essentially the initial condition problem in ordinary differential equation (ODE). The solution trajectory of an ODE is determined by its initial condition. Therefore, different initial conditions will yield distinct solution trajectories. Hence, without i.i.c., it is impossible for the control system to generate a* *solution trajectory that is identical to the reference $x_d(t)$. That is the main reason why i.i.c. is indispensable for perfect tracking. For more discussion, please see Remark 2.4 in Chapter 2.*

From the tracking error definition in (6.2), we can find out the error dynamics as follows.

$$\begin{aligned}
\dot{e}_{i,j} &= \dot{x}_d - \dot{x}_{i,j}, \\
&= \dot{x}_d - \theta_j^0 \xi_j^0(t, x_{i,j}) - \eta_j(t, x_{i,j}) - b_j u_{i,j} \\
&= \dot{x}_d - \eta_j(t, x_d) - \theta_j^0 \xi_j^0(t, x_{i,j}) - b_j u_{i,j} + (\eta_j(t, x_d) - \eta_j(t, x_{i,j})) \\
&= b_j \theta_j \xi_j(t, x_{i,j}) + (\eta_j(t, x_d) - \eta_j(t, x_{i,j})) - b_j u_{i,j},
\end{aligned} \tag{6.6}$$

where

$$\theta_j \triangleq [-b_j^{-1}\theta_j^0,\ b_j^{-1}(\dot{x}_d - \eta_j(t, x_d))] \in \mathbb{R}^{1\times(n_j+1)}, \tag{6.7}$$

$$\xi_j(t, x_{i,j}) \triangleq [(\xi_j^0(t, x_{i,j}))^T,\ 1]^T \in \mathbb{R}^{(n_j+1)\times 1}. \tag{6.8}$$

Note that since $b_j \neq 0$, (6.7) and (6.8) are well-defined.

The error dynamics (6.6) contain parametric uncertainty $\theta_j^0 \xi_j^0(t, x_{i,j})$, the lumped uncertainty $\eta_j(t, x_{i,j})$, and the derivative of unknown desired trajectory $\dot{x}_d$. Equation (6.6) shows by lumping $\dot{x}_d - \eta_j(t, x_d)$ into parametric uncertainties, it is possible to incorporate adaptive ILC with robust ILC to deal with parametric uncertainties $\theta_j \xi_j(t, x_{i,j})$ and $\eta_j(t, x_d) - \eta_j(t, x_{i,j})$ with the help of Assumption 6.1, provided that $e_{i,j}$ is available for each follower. However, $e_{i,j}$ cannot be used to design ILC schemes because only a small portion of followers can access the leader's trajectory. Remark 6.3 shows by using local information, consensus tracking can be achieved when an ILC law is designed appropriately.

6.3 Controller Design for First-order Multi-agent Systems

This section contains two parts. The first part shows how to design ILC law to ensure robust consensus tracking for the multi-agent systems discussed in Section 6.2.2. The second part discusses how to design ILC updating law for system (6.1) if Assumption 6.3 is relaxed.

6.3.1 Main results

In the following convergence analysis, let Δ represent the difference operator over two consecutive iterations. Specifically, $\Delta\omega_i := \omega_i - \omega_{i-1}$, where $\omega_i \in \{E_i, V_i\}$, and E_i, V_i are defined later.

Obviously, the proposed ILC updating law for the jth agent needs to ensure the consensus tracking, reject lumped uncertainties, and learn parametric uncertainties. Therefore, it has three components:

$$u_{i,j} = \underbrace{\underline{b}_j^{-1}\gamma\varepsilon_{i,j}}_{\text{consensus tracking}} + \underbrace{\underline{b}_j^{-1}\left(\overline{\phi}_j(x_{i,j})\right)^2\varepsilon_{i,j}}_{\text{robust control}} + \underbrace{\hat{\theta}_{i,j}\xi_j(t,x_{i,j})}_{\text{parameter learning}}, \tag{6.9}$$

where

$$\overline{\phi}_j(x_{i,j}) = \sup_{|z|\le b_d} \phi_j(z,x_{i,j}), \tag{6.10}$$

γ is a positive constant to be designed, and $\hat{\theta}_{i,j}$ is an estimation of θ_j at the ith iteration. It is updated as follows

$$\begin{aligned} \hat{\theta}_{i,j}(t) &= \hat{\theta}_{i-1,j}(t) + \kappa\varepsilon_{i,j}\left(\xi_j(t,x_{i,j})\right)^T, \\ \hat{\theta}_{0,j}(t) &= 0, \quad \forall t\in[0,T], \end{aligned} \tag{6.11}$$

where $\kappa > 0$ is the parameter learning gain.

The updating laws (6.9) can be rewritten in a stack vector form,

$$\mathbf{u}_i = \underline{B}^{-1}\left(\gamma I + \left(\overline{\Phi}(\mathbf{x}_i)\right)^2\right)\varepsilon_i + \hat{\Theta}_i\xi(t,\mathbf{x}_i), \tag{6.12}$$

where $\underline{B} \triangleq \mathrm{diag}(\underline{b}_1,\ldots,\underline{b}_N)$; ε_i, $\mathbf{u}_i$ and $\xi(t,\mathbf{x}_i)$ are the column stack vectors of $\varepsilon_{i,j}$, $u_{i,j}$ and $\xi_j(t,x_{i,j})$ respectively, and

$$\begin{aligned}
\overline{\Phi}(\mathbf{x}_i) &\triangleq \mathrm{diag}(\overline{\phi}_1(x_{i,1}),\overline{\phi}_2(x_{i,2})\ldots,\overline{\phi}_N(x_{i,N})),\\
\hat{\Theta}_i &\triangleq \mathrm{diag}(\hat{\theta}_{i,1},\hat{\theta}_{i,2},\ldots,\hat{\theta}_{i,N}).
\end{aligned}$$

For convenience, we introduce the following notations,

$$\begin{aligned}
\eta_d &\triangleq \begin{bmatrix} \eta_1(t,x_d), & \eta_2(t,x_d), & \cdots, & \eta_N(t,x_d) \end{bmatrix}^T,\\
\eta(t,\mathbf{x}_i) &\triangleq \begin{bmatrix} \eta_1(t,x_{i,1}), & \eta_2(t,x_{i,2}), & \cdots, & \eta_N(t,x_{i,N}) \end{bmatrix}^T,\\
B &\triangleq \mathrm{diag}(b_1,b_2\ldots,b_N),\\
\Theta &\triangleq \mathrm{diag}(\theta_1,\theta_2,\ldots,\theta_N),\\
\tilde{\Theta}_i &\triangleq \Theta - \hat{\Theta}_i.
\end{aligned}$$

This leads to the following closed loop error dynamics,

$$\dot{\mathbf{e}}_i = (\eta_d - \eta(t,\mathbf{x}_i)) + B\underline{B}^{-1}\left(\gamma I + \left(\overline{\Phi}(\mathbf{x}_i)\right)^2\right)\varepsilon_i + \tilde{\Theta}_i\xi(t,\mathbf{x}_i). \tag{6.13}$$

The following CEF is introduced:

$$E_i(t) = \underbrace{V_i(\mathbf{e}_i)}_{\text{consensus tracking}} + \underbrace{\frac{1}{2\kappa}\int_0^t \mathrm{Trace}\left(\left(\tilde{\Theta}_i(\tau)\right)^T B\tilde{\Theta}_i(\tau)\right)d\tau}_{\text{parameter learning}}, \tag{6.14}$$

where $V_i(\mathbf{e}_i) \triangleq \frac{1}{2}\mathbf{e}_i^T H\mathbf{e}_i$. The CEF contains the information related to consensus tracking and the parameter learning. It is defined for each time instant over $[0,T]$.

The first result is stated in Theorem 6.1.

Theorem 6.1 *Assume that Assumptions 6.1–6.3 hold for the multi-agent system (6.1). The closed loop system consisting of (6.1) and the updating laws (6.9)–(6.11), can ensure that the tracking error $e_{i,j}(t)$ converges to zero point-wisely ($j = 1,2,\ldots,N$) for any $t \in [0,T]$ along the iteration axis, if*

$$\gamma \geq \frac{1}{4\underline{\sigma}(H^2)} + \alpha, \tag{6.15}$$

for some positive constant α. Moreover, $u_{i,j} \in \mathscr{L}^2[0,T]$ for any $j = 1,2,\ldots,N$, $i \in \mathbb{N}_{\geq 0}$.

Proof: see Appendix B.3. ■

Remark 6.7 *In the proof of Theorem 6.1, it shows that $E_{i+1}(t) \leq E_i(t) - V_i(\mathbf{e}_i(t))$ for each time instant over $[0,T]$. After showing the uniform boundedness of $E_1(t)$ over time, the point-wise convergence of the tracking error can be ensured. The uniform boundedness of $E_i(t)$ for any $i \in \mathbb{N}_{\geq 0}$, $t \in [0,T]$ shows that the control input is $\mathscr{L}^2$ bounded uniformly for all $i \in \mathbb{N}_{\geq 0}$.*

Remark 6.8 *As pointed out by Xu and Tan (2002a), the parameter updating law (6.11) cannot ensure the uniform boundedness of the $\hat{\Theta}_i, i \in \mathbb{N}_{\geq 0}$, only the point-wise convergence of the tracking error can be ensured and the input signal is $\mathscr{L}^2$ bounded. When an appropriate projection method is used for the updating law (6.11), it is possible to get stronger results (uniform convergence of the tracking error and uniform boundedness of the input signal). However, the projection method always requires extra information: the upper and lower bounds of unknown time-varying parameters.*

Remark 6.9 *In the parameter updating rule (6.11), the derivative of the desired trajectory $\dot{x}_d$ is treated as a completely unknown variable, though, some of the followers are able to know $\dot{x}_d$. This available information can be used to possibly improve the performance (for example, transient response along iteration domain). Let Ω_0 be the* *set of agents who can obtain the leader's information, i.e.,* $\Omega_0 \triangleq \{j \in \mathscr{V} | d_j = 1\}$*, which leads to another parameterization:*

$$\theta_j^a \triangleq \begin{cases} [-b_j^{-1}\theta_j^0, -b_j^{-1}\eta_j(t,x_d), b_j^{-1}], & \text{if } j \in \Omega_0, \\ [-b_j^{-1}\theta_j^0, b_j^{-1}(\dot{x}_d - \eta_j(t,x_d))], & \text{otherwise}, \end{cases}$$

and

$$\xi_j^a(t,x_{i,j}) \triangleq \begin{cases} \left[(\xi_j^0(t,x_{i,j}))^T, 1, \dot{x}_d\right]^T, & \text{if } j \in \Omega_0, \\ [(\xi_{i,j}^0(t,x_{i,j}))^T, 1]^T, & \text{otherwise}. \end{cases}$$

Under this parameterization, the parameter updating law (6.11) is applicable if θ_j *and* ξ_j *are replaced by* θ_j^a *and* ξ_j^a *respectively. It is interesting to observe when more information is available, the number of unknown parameters increases, leading to a more complicated controller.*

6.3.2 Extension to alignment condition

Although Assumption 6.3 is a necessary condition to ensure the perfect tracking performance, it is not easy to achieve *i.i.c.* in general. It is widely observed in the industry that many motion systems start from the position where they stopped in the previous iteration (Xu and Qu, 1998; Xu and Xu, 2004). For example, consider the industry manipulator performing the pick and place task repetitively. The starting position is always the final position in the previous task execution. This motivates the widely used alignment condition in the area of ILC.

Assumption 6.4 *For any* $i \in \mathbb{N}_{\geq 0}$ *and* $j = 1, 2, \ldots, N$*, the system (6.1) satisfies* $x_{i+1,j}(0) = x_{i,j}(T)$*. Moreover, the desired trajectory also satisfies* $x_d(0) = x_d(T)$*.*

With this relaxed assumption, we can obtain the following weaker result.

Corollary 6.1 *Assume that Assumptions 6.1, 6.2 and 6.4 hold for the multi-agent system (6.1). The closed loop system consisting of (6.1) and the updating laws (6.9)–(6.11)* *satisfying (6.15), can ensure that the tracking error* $e_{i,j}(t)$ *converges in the sense of* $\mathscr{L}^2[0,T]$ *norm, i.e.,*

$$\lim_{i\to\infty}\int_0^T (e_{i,j}(\tau))^2\,d\tau = 0. \tag{6.16}$$

Moreover, $u_{i,j}\in\mathscr{L}^2[0,T]$ *for any* $j=1,2,\ldots,N$, $i\in\mathbb{N}_{\geq 0}$

Proof: see Appendix B.4. ■

Remark 6.10 *As Assumption 6.3 is relaxed by Assumption 6.4, Corollary 6.1 obtains a weaker convergent result compared with that in Theorem 6.1, namely, from point-wise to* $\mathscr{L}^2$ *convergence.*

6.4 Extension to High-order Systems

With some nontrivial modifications of the proposed methods in Section 6.3, the learning controllers can be applied to high-order systems. The results are first derived under *i.i.c.*, then extended to the imperfect initial conditions with the inital rectifying action.

Consider the jth agent modeled by the following dynamics,

$$\begin{aligned}\dot{x}_{i,j_k} &= x_{i,j_{k+1}},\, k=1,2,\ldots,n-1,\\ \dot{x}_{i,j_n} &= \theta_j^0\xi_j^0(t,\mathbf{x}_{i,j})+\eta_j(t,\mathbf{x}_{i,j})+b_j(t)u_{i,j},\end{aligned} \tag{6.17}$$

where $\mathbf{x}_{i,j}$ is the column stack vector of x_{i,j_k}, $k=1,\ldots,n$.

The desired trajectory $\mathbf{x}_d=[x_{d_1},x_{d_2},\ldots,x_{d_n}]^T$ is generated by

$$\begin{aligned}\dot{x}_{d_k} &= x_{d_{k+1}},\, k=1,2,\ldots,n-1,\\ \dot{x}_{d_n} &= \dot{x}_d.\end{aligned}$$

Define the following two auxiliary variables:

$$s_{i,j} \triangleq \sum_{k=1}^{n} c_k x_{i,j_k}, \text{ and } s_d \triangleq \sum_{k=1}^{n} c_k x_{d_k},$$

where $c_n = 1$, and

$$\lambda^{n-1} + c_{n-1}\lambda^{n-2} + \cdots + c_1 = 0 \tag{6.18}$$

is a stable polynomial. For instance, c_i can be chosen the same coefficient as the polynomial $(\lambda+1)^{n-1}$.

Let the tracking error for the jth agent at the ith iteration be $e_{i,j} = s_d - s_{i,j}$. Since the the polynomial (6.18) is stable, and $\mathbf{x}_{i,j}(0) = \mathbf{x}_d(0)$, hence, if $e_{i,j} = 0$, perfect tracking is achieved, i.e., $\mathbf{x}_{i,j}(t) = \mathbf{x}_d(t), t \in [0,T]$. The corresponding extended tracking error is defined as

$$\varepsilon_{i,j} \triangleq \sum_{k \in \mathscr{N}_j} a_{j,k}(s_{i,k} - s_{i,j}) + d_j(s_d - s_{i,j}). \tag{6.19}$$

Similarly, the extended tracking error $\varepsilon_{i,j}$ in (6.19) can be written in the compact matrix form,

$$\varepsilon_i = -L\mathbf{s}_i + D\mathbf{e}_i = H\mathbf{e}_i, \tag{6.20}$$

where ε_i, $\mathbf{e}_i$, and $\mathbf{s}_i$ are the column stack vectors of $\varepsilon_{i,j}$, $e_{i,j}$, and $s_{i,j}$. Notice that the definitions of $e_{i,j}$ and $s_{i,j}$ are rather different from the ones in Section 6.3.

The error dynamics now become

$$\begin{aligned}
\dot{e}_{i,j} &= \sum_{k=1}^{n} c_k \dot{x}_{d_k} - \sum_{k=1}^{n} c_k \dot{x}_{i,j_k} \\
&= \sum_{k=1}^{n-1} c_k (x_{d_{k+1}} - x_{i,j_{k+1}}) + \dot{x}_d - \theta_j^0 \xi_j^0(t, \mathbf{x}_{i,j}) - \eta_j(t, \mathbf{x}_{i,j}) - b_j u_{i,j} \\
&= b_j \theta_j \xi_j(t, \mathbf{x}_{i,j}) + \eta_j(t, \mathbf{x}_d) - \eta_j(t, \mathbf{x}_{i,j}) - b_j u_{i,j},
\end{aligned} \tag{6.21}$$

where

$$\theta_j \triangleq \left[-b_j^{-1}\theta_j^0, b_j^{-1}\left(\sum_{k=1}^{n-1} c_k x_{d_{k+1}} + \dot{x}_d - \eta_j(t, \mathbf{x}_d) \right), -b_j^{-1} \right],$$

and

$$\xi_j(t,\mathbf{x}_{i,j}) \triangleq \left[\left(\xi_j^0(t,\mathbf{x}_{i,j})\right)^T, 1, \sum_{k=1}^{n-1} c_k x_{i,j_{k+1}}\right]^T.$$

The proposed ILC controller and updating rule are

$$u_{i,j} = \underline{b}_j^{-1}\left(\gamma + \left(\overline{\phi}_j(\mathbf{x}_{i,j})\right)^2\right)\varepsilon_{i,j} + \hat{\theta}_{i,j}\xi_j(t,\mathbf{x}_{i,j}), \tag{6.22}$$

$$\hat{\theta}_{i,j} = \hat{\theta}_{i-1,j} + \kappa\varepsilon_{i,j}\left(\xi_j(t,\mathbf{x}_{i,j})\right)^T, \hat{\theta}_{0,j} = 0, \tag{6.23}$$

where $\overline{\phi}_j(\mathbf{x}_{i,j}) = \sup_{|\mathbf{z}|\leq b_d} \phi_j(\mathbf{z},\mathbf{x}_{i,j})$, and $\hat{\theta}_{i,j}$ is the estimate of θ_j at the ith iteration.

To analyze the convergence properties of the proposed algorithm. The following result is required. Denote $\delta\mathbf{x}_{i,j} = \mathbf{x}_d - \mathbf{x}_{i,j}$, and $\mathbf{c}^T = [c_1,\ldots,c_n]$. Then we have $e_{i,j} = \mathbf{c}^T\delta\mathbf{x}_{i,j}$.

Lemma 6.2 *Under Assumption 6.3,* $\mathbf{x}_{i,j}(0) = \mathbf{x}_d(0)$*, we have*

$$|\delta\mathbf{x}_{i,j}| \leq r|e_{i,j}|,$$

where $r = \|e^{Ct}\mathbf{g}\|$, $C = \begin{bmatrix} 0 & 1 & 0 & \cdots & \cdots \\ 0 & 0 & 1 & 0 & \cdots \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & -c_1 & -c_2 & \cdots & -c_{n-1} \end{bmatrix}$, *and* $\mathbf{g} = \begin{bmatrix} 0 \\ 0 \\ \vdots \\ 1 \end{bmatrix}$.

Proof: Taking derivative of $e_{i,j}$ yields $\dot{e}_{i,j} = \mathbf{c}^T\delta\dot{\mathbf{x}}_{i,j}$. Rewrite $\delta\dot{\mathbf{x}}_{i,j}$ in the linear matrix form, one has

$$\delta\dot{\mathbf{x}}_{i,j} = C\delta\mathbf{x}_{i,j} + \mathbf{g}\dot{e}_{i,j}.$$

Notice Assumption 6.3 that $\delta\mathbf{x}_{i,j}(0) = 0$, the above differential equation renders the following solution,

$$\begin{aligned} \delta\mathbf{x}_{i,j} &= e^{Ct}\delta\mathbf{x}_{i,j}(0) + \int_0^t e^{C(t-\tau)}\mathbf{g}\dot{e}_{i,j}(\tau)\,d\tau \\ &= \int_0^t e^{C(t-\tau)}\mathbf{g}\dot{e}_{i,j}(\tau)\,d\tau. \end{aligned}$$

Taking norm operation from the both sides yields,

$$|\delta \mathbf{x}_{i,j}| \leq \left(\max_{t\in[0,T]} |e^{Ct}\mathbf{g}| \right) \left| \int_0^t \dot{e}_{i,j}(\tau)\, d\tau \right| = r|e_{i,j}|.$$

■

Remark 6.11 *The characteristic polynomial of C is $\lambda(\lambda^{n-1} + c_{n-1}\lambda^{n-2} + \ldots + c_1) = 0$. Notice that the polynomial in (6.18) is stable, therefore, zero is one of the eigenvalues of C, and all the rest eigenvalues are stable. Hence, $\|e^{Ct}\|$ is finite for all $t \in [0, \infty)$.*

Define

$$\begin{aligned} \eta(t, \mathbf{x}_d) &\triangleq [\eta_1(t, \mathbf{x}_d), \eta_2(t, \mathbf{x}_d), \ldots, \eta_N(t, \mathbf{x}_d)]^T, \\ \eta(t, \mathbf{x}_i) &\triangleq [\eta_1(t, \mathbf{x}_{i,1}), \eta_2(t, \mathbf{x}_{i,2}), \ldots, \eta_N(t, \mathbf{x}_{i,N})]^T. \end{aligned}$$

With the help of Lemma 6.2, the equation (B.9) in Appendix B.3 for the first-order case now becomes

$$|\varepsilon_i^T(\eta(t, \mathbf{x}_d) - \eta(t, \mathbf{x}_i))| \leq \varepsilon_i^T \left(\overline{\Phi}(\mathbf{x}_i)\right)^2 \varepsilon_i + \frac{1}{4\underline{\sigma}(H)^2} r^2 \varepsilon_i^T \varepsilon_i. \tag{6.24}$$

Theorem 6.2 *Assume that Assumptions 6.1–6.3 hold for the high-order multi-agent system (6.17). The closed loop system consisting of (6.17) and the updating rules (6.22)–(6.23), can ensure that the tracking error $e_{i,j}(t)$ approaches to zero point-wisely ($j = 1, 2, \ldots, N$) for any $t \in [0, T]$ along iteration axis, if*

$$\gamma \geq \frac{r^2}{4\underline{\sigma}(H^2)} + \alpha,$$

for some positive constant α. Moreover, $u_{i,j} \in \mathscr{L}^2[0, T]$ for any $j = 1, 2, \ldots, N$, $i \in \mathbb{N}_{\geq 0}$.

Sketch of proof: The error dynamics (6.21) and controllers (6.22), (6.23) are very similar to the ones in the first-order case, except that the parameterizations become more complex and the error term definitions are different. Consider the CEF below

$$E_i(t) = V_i(\mathbf{e}_i) + \frac{1}{2\kappa} \int_0^t \mathrm{Trace}\left(\tilde{\Theta}_i^T B \tilde{\Theta}_i\right) d\tau,$$

together with (6.24), follow the similar proof procedures in the previous section, we conclude Theorem 6.2. ■

Theorem 6.2 depends on the *i.i.c.*, which makes the results sensitive to the initial conditions. Assume that the initial state of each agent cannot be manipulated and is reset to a fixed state at every iteration, but not the same as the desired state $\mathbf{x}_d(0)$. By extending the concept of the initial rectifying action (Sun and Wang, 2002), it is possible to make the results more robust to the CEF based ILC.

Assumption 6.5 *The initial state of the jth follower is reset to a fixed initial state at every iteration, i.e.,* $\mathbf{x}_{i,j}(0) = \mathbf{x}_j(0) \neq \mathbf{x}_d(0)$ *for all* $i \in \mathbb{N}_{\geq 0}$.

Define matrices $A, \bar{A} \in \mathbb{R}^{n\times n}$ below,

$$A = \begin{bmatrix} 0 & 1 & 0 & \cdots & 0 \\ 0 & 0 & 1 & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & \cdots & \cdots & \cdots & 0 \end{bmatrix},$$

and $\bar{A} = A - \mathbf{g}K$ is a stable matrix, where $K \in \mathbb{R}^{1\times n}$ and $\mathbf{g}$ is defined in Lemma 6.2. K can be interpreted as the state feedback gain. The structure of A is decided by the structure of system model (6.17). Let the extended tracking error be

$$\begin{aligned} \hat{\varepsilon}_{i,j} \triangleq & \sum_{k\in\mathcal{N}_j} a_{j,k}(s_{i,k} - s_{i,j}) + d_j(s_d - s_{i,j}) - \sum_{k\in\mathcal{N}_j} a_{j,k}\mathbf{c}^T e^{\bar{A}t}(\mathbf{x}_k(0) - \mathbf{x}_j(0)) \\ & - d_j \mathbf{c}^T e^{\bar{A}t}(\mathbf{x}_d(0) - \mathbf{x}_j(0)). \end{aligned} \tag{6.25}$$

It is worth noting that the extended tracking error in (6.25) is a distributed measurement. Hence, it can be used in the distributed controller design.

To design and analyze the distributed version of initial rectifying action in the CEF framework, define the following auxiliary variables:

$$\begin{aligned}\hat{\mathbf{x}}_{d,j} &\triangleq \mathbf{x}_d - e^{\bar{A}t}(\mathbf{x}_d(0)-\mathbf{x}_j(0)),\\ \hat{s}_{d,j} &\triangleq \mathbf{c}^T\hat{\mathbf{x}}_{d,j} = s_d - \mathbf{c}^T e^{\bar{A}t}(\mathbf{x}_d(0)-\mathbf{x}_j(0)),\\ \hat{e}_{i,j} &\triangleq \hat{s}_{d,j} - s_{i,j}.\end{aligned}$$

The dynamics of $\hat{e}_{i,j}$ can be obtained as

$$\begin{aligned}\dot{\hat{e}}_{i,j} &= \dot{\hat{s}}_{d,j} - \dot{s}_{i,j}\\ &= \dot{\hat{s}}_{d,j} - \sum_{k=1}^{n-1} c_k x_{i,j_{k+1}} - \theta_j^0\xi_j^0(t,\mathbf{x}_{i,j}) - \eta_j(t,\mathbf{x}_{i,j}) - b_j u_{i,j}\\ &= b_j\theta_j\xi_j(t,\mathbf{x}_{i,j}) + \eta_j(t,\hat{\mathbf{x}}_{d,j}) - \eta_j(t,\mathbf{x}_{i,j}) - b_j u_{i,j},\end{aligned} \tag{6.26}$$

where the new parameterizations are

$$\theta_j \triangleq \left[-b_j^{-1}\theta_j^0,\, b_j^{-1}\left(\dot{\hat{s}}_{d,j} - \eta_j(t,\hat{\mathbf{x}}_{d,j})\right),\, -b_j^{-1}\right],$$

and

$$\xi_j(t,\mathbf{x}_{i,j}) \triangleq \left[\left(\xi_j^0(t,\mathbf{x}_{i,j})\right)^T,\, 1,\, \sum_{k=1}^{n-1} c_k x_{i,j_{k+1}}\right]^T.$$

The dynamics in (6.26) are identical to (6.21) except that the error definition and parameterizations are different. Analogous to the controller design in the *i.i.c.* case, the proposed ILC controller and updating rule are

$$u_{i,j} = \underline{b}_j^{-1}\left(\gamma + \left(\overline{\phi}_j(\mathbf{x}_{i,j})\right)^2\right)\hat{\varepsilon}_{i,j} + \hat{\theta}_{i,j}\xi_j(t,\mathbf{x}_{i,j}), \tag{6.27}$$

$$\hat{\theta}_{i,j} = \hat{\theta}_{i-1,j} + \kappa\hat{\varepsilon}_{i,j}\left(\xi_j(t,\mathbf{x}_{i,j})\right)^T,\ \hat{\theta}_{0,j} = 0, \tag{6.28}$$

where $\overline{\phi}_j(\mathbf{x}_{i,j}) = \sup_{|\mathbf{z}|\le\hat{b}_d} \phi_j(\mathbf{z},\mathbf{x}_{i,j})$, $\hat{b}_d = b_d + |\mathbf{x}_d(0) - \mathbf{x}_j(0)|$, and $\hat{\theta}_{i,j}$ is the estimate of θ_j at the ith iteration.

To analyze the convergence properties, $\hat{\varepsilon}_{i,j}$ should be expressed in terms of $\hat{e}_{i,j}$. With the help of auxiliary variables, we have

$$\begin{aligned}
\hat{\varepsilon}_{i,j} &= \sum_{k\in\mathscr{N}_j} a_{j,k}(\hat{s}_{d,j} - s_{i,j} - (\hat{s}_{d,k} - s_{i,k}) - \hat{s}_{d,j} + \hat{s}_{d,k}) - \sum_{k\in\mathscr{N}_j} a_{j,k}\mathbf{c}^T e^{\bar{A}t}(\mathbf{x}_k(0) - \mathbf{x}_j(0)) \\
&\quad + d_j(s_d - s_{i,j}) - d_j\mathbf{c}^T e^{\bar{A}t}(\mathbf{x}_d(0) - \mathbf{x}_j(0)) \\
&= \sum_{k\in\mathscr{N}_j} a_{j,k}(\hat{e}_{i,j} - \hat{e}_{i,k}) + d_j\hat{e}_{i,j}.
\end{aligned}$$

Rewriting the above equation in compact matrix form yields

$$\hat{\varepsilon}_i = H\hat{\mathbf{e}}_i, \tag{6.29}$$

where $\hat{\varepsilon}_i$ and $\hat{\mathbf{e}}_i$ are the column stack vectors of $\hat{\varepsilon}_{i,j}$ and $\hat{e}_{i,j}$ respectively.

Based on the definition of $\hat{\mathbf{x}}_{d,j}$, it is guaranteed that $\hat{\mathbf{x}}_{d,j}(0) = \mathbf{x}_j(0)$. Define $\delta\hat{\mathbf{x}}_{i,j} = \hat{\mathbf{x}}_{d,j} - \mathbf{x}_{i,j}$, following the idea in Lemma 6.2, we have $|\delta\hat{\mathbf{x}}_{i,j}| \leq r|\hat{e}_{i,j}|$.

With the above developments and analogy to Theorem 6.2, we have the following results.

Theorem 6.3 *Assume that Assumptions 6.1, 6.2, and 6.5 hold for the high-order multi-agent system (6.17). The closed loop system consisting of (6.17) and the updating rules (6.27)–(6.28), can ensure that the tracking error $\hat{e}_{i,j}(t)$ approaches to zero point-wisely ($j = 1, 2, \ldots, N$) for any $t \in [0, T]$ along iteration axis, if*

$$\gamma \geq \frac{r^2}{4\underline{\sigma}(H^2)} + \alpha,$$

for some positive constant α. Therefore,

$$\lim_{i\to\infty} \mathbf{x}_{i,j}(t) = \mathbf{x}_d(t) - e^{\bar{A}t}(\mathbf{x}_d(0) - \mathbf{x}_j(0)).$$

Moreover, $u_{i,j} \in \mathscr{L}^2[0, T]$ for any $j = 1, 2, \ldots, N$, $i \in \mathbb{N}_{\geq 0}$.

The convergence results can be proved analogously to Theorems 6.1 and 6.2 by studying the following CEF

$$E_i(t) = V_i(\hat{\mathbf{e}}_i) + \frac{1}{2\kappa}\int_0^t \mathrm{Trace}\left(\tilde{\Theta}_i^T B\tilde{\Theta}_i\right) d\tau.$$

As discussed in Remark 6.6, the perfect tracking can never be achieved when $i.i.c.$ is violated. From Theorem 6.3, it can be seen that the final trajectory of each individual agent can be adjusted by the matrix K in $\bar{A}$.

6.5 Illustrative Example

To illustrate the applications of the developed algorithms, consider a group of 4 agents. The communication graph among followers and leader is depicted in Figure. 6.1. Vertex 0 represents the virtual leader, and the dashed lines stand for the communication links betweens leader and followers, i.e., only agents 1 and 4 can access the state information of the leader. The solid lines denote the communication links between followers, and the communication graph among followers is connected. Then the Laplacian is

$$L = \begin{bmatrix} 2 & -1 & 0 & -1 \\ -1 & 3 & -1 & -1 \\ 0 & -1 & 1 & 0 \\ -1 & -1 & 0 & 2 \end{bmatrix},$$

and $D = \mathrm{diag}(1,0,0,1)$. Hence, the smallest singular value of H is $\underline{\sigma}(H) = 0.3249$.

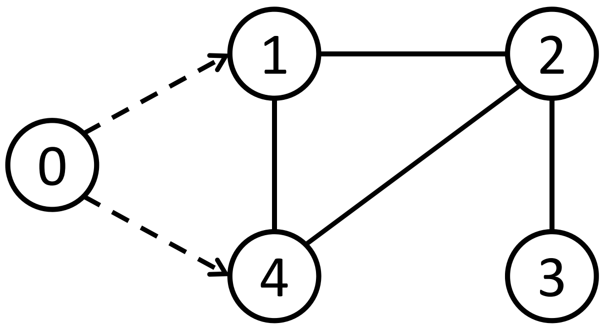

Figure 6.1: Communication among agents in the network.

In the simulation study, we first demonstrate the results for first-order systems under *i.i.c.* and alignment condition. Next, illustrate the results for high-order systems.

6.5.1 First-order Agents

Let the 4 followers be modeled by the following dynamics,

$$\begin{aligned}
\dot{x}_{i,1} &= \sin(t)x_{i,1}^2 + (1+0.1\sin^2(t))u_{i,1} + \eta_1(t,x_{i,1}),\\
\dot{x}_{i,2} &= \cos(t)x_{i,2}^3 + (1+0.2\sin^2(t))u_{i,2} + \eta_2(t,x_{i,2}),\\
\dot{x}_{i,3} &= e^{-t}x_{i,3}^2 + (1+0.3\sin^2(t))u_{i,3} + \eta_3(t,x_{i,3}),\\
\dot{x}_{i,4} &= -tx_{i,4}^3 + (1+0.4\sin^2(t))u_{i,4} + \eta_4(t,x_{i,4}),
\end{aligned}$$

where the unknown disturbance $\eta_j(t,x_{i,j}) = x_{i,j}^2\sin(j\cdot t)$ for $j=1,2,3,4$. Notice that the disturbance $\eta_j(t,x_{i,j})$ satisfies the Assumption 6.1, i.e.,

$$|\eta_j(t,x_d) - \eta_j(t,x_{i,j})| \leq \phi_j(x_d,x_{i,j})|x_d - x_{i,j}|,$$

where $\phi_j(x_d,x_{i,j}) = |x_d| + |x_{i,j}|$.

The agent dynamics have the same form as in (6.1). So the learning rules (6.9) and (6.11) can be applied directly.

Identical Initialization Condition

Choose the desired trajectory $x_d = \sin^3(t)$, $t \in [0,5]$, and let all the agents satisfy the *i.i.c.*, i.e., $x_{i,j}(0) = x_d(0)$. Set $\gamma = 7.4$, $\kappa = 5$, and $\overline{\phi}_j(x_{i,j}) = 1 + |x_{i,j}|$.

Figure. 6.2 shows the trajectory profiles at the 1st and 50th iterations. At the 1st iteration, the followers' trajectories have some deviations from the leader's. Due to the current error feedback in the controller, the transient response is much better than the typical contraction-mapping based controllers. The followers' trajectories almost overlap with the leader's at the 50th iteration. Figure. 6.3 shows the maximum tracking error

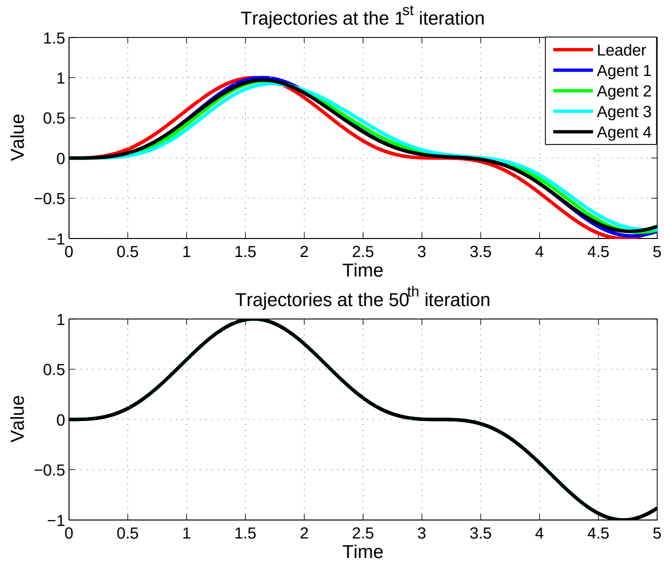

Figure 6.2: The trajectory profiles at the 1st and 50th iterations under *i.i.c.*

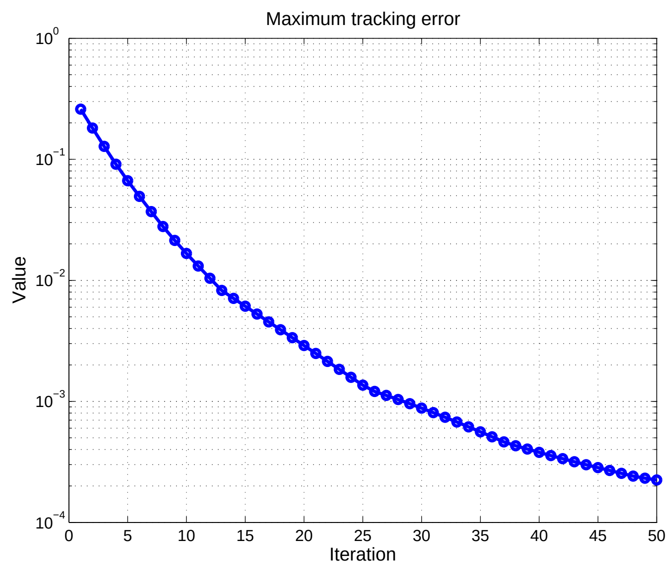

Figure 6.3: Maximum tracking error vs. iteration number under *i.i.c.*

versus iteration number. The maximum tracking error has been reduced to 0.0086% of the one at the 1st iteration, where the maximum tracking error at the ith iteration is defined as $\max_{j=1,2,3,4} \max_{t\in[0,T]} |x_d - x_{i,j}|$.

Alignment Condition

Choose the desired trajectory $x_d = \sin^3(t), t \in [0, \pi]$. x_d is a closed orbit since $x_d(0) = x_d(\pi)$. The initial condition for the follower agents are $x_{1,1}(0) = 0.2$, $x_{1,2}(0) = 0.4$, $x_{1,3}(0) = 0.6$, $x_{1,4}(0) = -0.4$. Set $\gamma = 7.4$, $\kappa = 5$, and $\overline{\phi}_j(x_{i,j}) = 1 + |x_{i,j}|$.

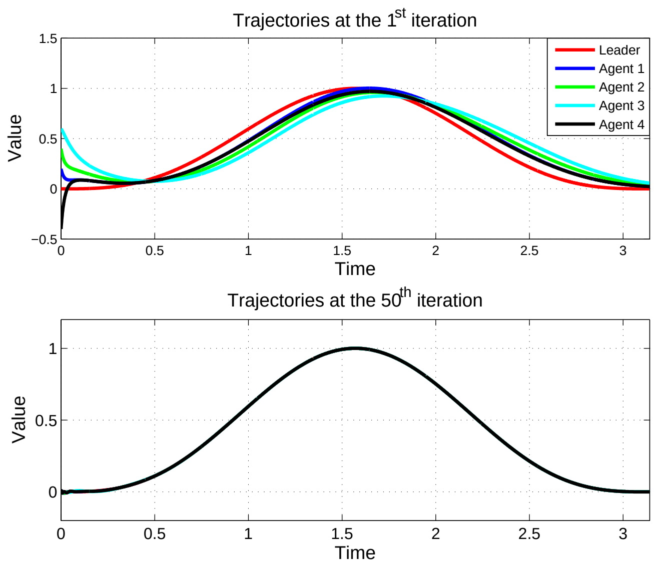

Figure 6.4: The trajectory profiles at the 1st and 50th iterations under alignment condition.

Figure. 6.4 shows the trajectory profiles at the 1st and 50th iterations. At the 1st iteration, the followers' trajectories have large deviations from the leader's, especially at the starting time. Due to the alignment condition, the controllers (6.9) and (6.11) can still work. The trajectories almost overlap with the leader's at the 50th iteration. Figure. 6.5 shows the maximum tracking error versus iteration number. The maximum

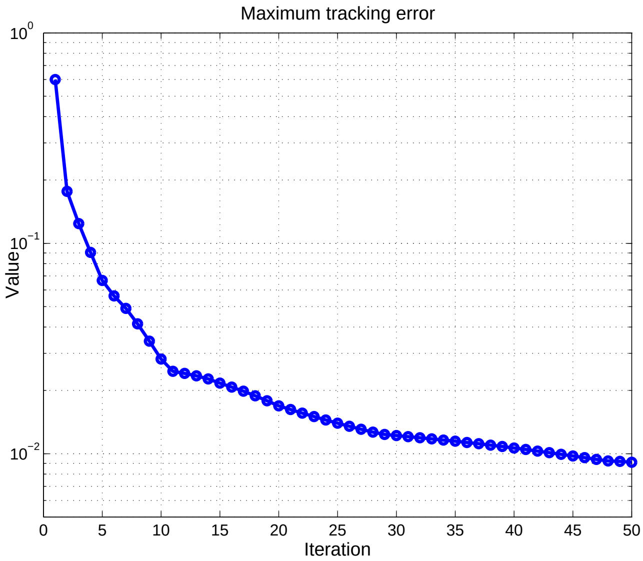

Figure 6.5: Maximum tracking error vs. iteration number under alignment condition.

tracking error has been reduced to 1.52% of the one at the 1st iteration.

6.5.2 High-order Agents

Consider the agent dynamics as follows,

$$\begin{bmatrix} \dot{x}_{i,j_1} \\ \dot{x}_{i,j_2} \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} x_{i,j_1} \\ x_{i,j_2} \end{bmatrix} + \begin{bmatrix} 0 \\ \frac{1+0.1*j\sin^2 t}{m_j l_j^2 + I_j} \end{bmatrix} \begin{bmatrix} 0 \\ u_{i,j} - g l_j \cos(x_{i,j_1}) \end{bmatrix} + \begin{bmatrix} 0 \\ \eta_j(t, \mathbf{x}_{i,j}) \end{bmatrix}, \tag{6.30}$$

where x_{i,j_1} is the position, x_{i,j_2} is the velocity, m_j is the mass, l_j is the length of the rigid-body, I_j is the moment of inertia, $u_{i,j}$ is the control input, and $\eta_j(t,\mathbf{x}_{i,j}) = x_{i,j_1}^2 \sin(j \cdot t)$ is the disturbance. The plant parameters are assumed to be unknown. In the simulation example, they are specified in Table. 6.1.

The agent dynamics are slightly different from the system in (6.17), with some

Table 6.1: Agent Parameters.

Agent	$m\,(kg)$	$l\,(m)$	$I\,(kg\cdot m^2)$	b
1	1.5	0.8	0.48	$0.60(1+.1\sin^2 t)$
2	2	0.9	0.81	$0.38(1+.2\sin^2 t)$
3	1.8	1	0.9	$0.37(1+.3\sin^2 t)$
4	1.7	0.7	0.42	$0.62(1+.4\sin^2 t)$

modifications of the parameterization, controllers (6.22) and (6.23) are still applicable. In this part of simulation study, the *i.i.c.* is assumed. The leader's trajectory is chosen as $x_{d_1}(t)=\sin^3 t$ for $t\in[0,5]$, and the controller parameters are chosen as $c=0.5$, $\gamma=10$, $\kappa=15$, and $\overline{\phi}_j(\mathbf{x}_{i,j})=1+|x_{i,j_1}|$.

Let $\mathbf{c}=[c,\,1]^T$, and $c=0.5$. The auxiliary variable $s_{i,j}=cx_{i,j_1}+x_{i,j_2}$, and $s_d=cx_{d_1}+x_{d_2}$. Then, the error dynamics are

$$\begin{aligned}\dot{e}_{i,j} &= cx_{d_2}-cx_{i,j_1}+\dot{x}_d-\eta_j(t,\mathbf{x}_{i,j})-b_j(u_{i,j}-gl_jcos(x_{i,j_1}))\\ &= b_j\theta_j\xi_{i,j}+\eta_j(t,\mathbf{x}_d)-\eta_j(t,\mathbf{x}_{i,j})-b_ju_{i,j},\end{aligned}$$

where

$$\theta_j\triangleq[gl_j,\,b_j^{-1}(cx_{d_2}+\dot{x}_d-\eta_j(t,\mathbf{x}_d)),\,-b_j^{-1}],$$

and

$$\xi_{i,j}\triangleq[cos(x_{i,j_1}),\,1,\,cx_{i,j_2}]^T.$$

From Lemma 6.2, it can be shown that $r=\max_{t\in[0,5]}|e^{Ct}\mathbf{g}|<2$, where $C=\begin{bmatrix}0 & 1\\ 0 & -.5\end{bmatrix}$, and $\mathbf{g}=\begin{bmatrix}0\\ 1\end{bmatrix}$.

In the simulation study, the following control laws are applied,

$$u_{i,j} = \underline{b}_j^{-1}(\gamma + (\overline{\phi}_j(\mathbf{x}_{i,j}))^2)\varepsilon_{i,j} + \hat{\theta}_{i,j}\xi_j(t, \mathbf{x}_{i,j}),$$

$$\hat{\theta}_{i,j} = \hat{\theta}_{i-1,j} + \kappa\varepsilon_{i,j}\left(\xi_j(t, \mathbf{x}_{i,j})\right)^T, \hat{\theta}_{0,j}(t) = 0,$$

where $\gamma = 10$, $\kappa = 15$, and $\overline{\phi}_j(\mathbf{x}_{i,j}) = 1 + |x_{i,j_1}|$. $5 \leq \hat{\theta}_{i,j_1} \leq 10$, $-20 \leq \hat{\theta}_{i,j_1} \leq 20$, and $-4 \leq \hat{\theta}_{i,j_3} \leq -1$.

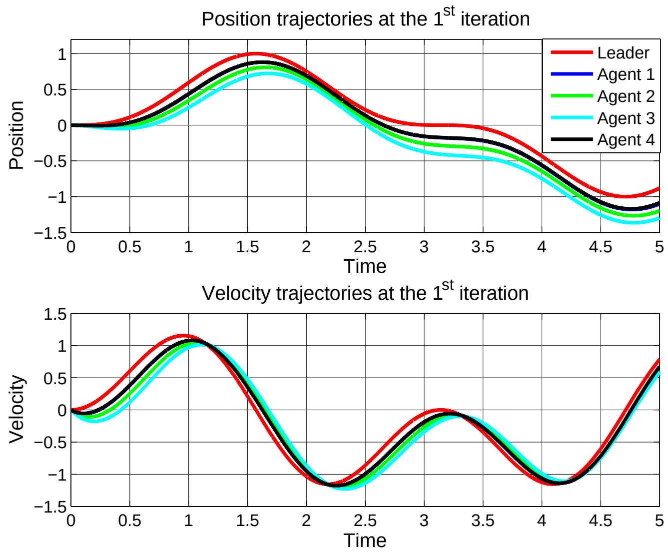

Figure 6.6: The trajectory profiles at the 1st iteration.

Under *i.i.c.*, Figures. 6.6 and 6.7 describe the position and velocity trajectories of all agents at the 1st and 50th iterations. At the 1st iteration, both positions and velocities of the followers do not match the leader's. At the 50th iteration the followers' trajectories overlap with the leader's.

Define the maximum position error at the ith iteration as $\max_{j=1,2,\ldots,4} \|x_{d_1} - x_{i,j_1}\|$, and the maximum velocity error is defined analogously with the position error. The maximum position and velocity tracking error profiles are shown in Figure. 6.8. Based on the simulation results, the maximum position tracking error at the 50th iteration has been reduced to 0.05% of the one at the 1st iteration, meanwhile, the corresponding

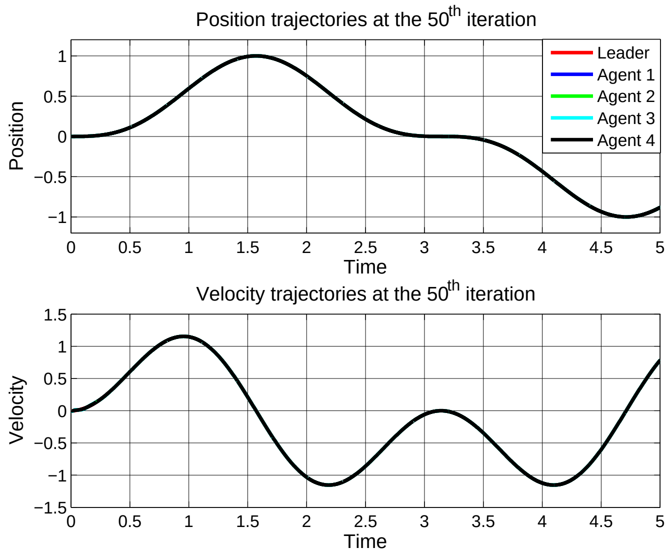

Figure 6.7: The trajectory profiles at the 50th iteration.

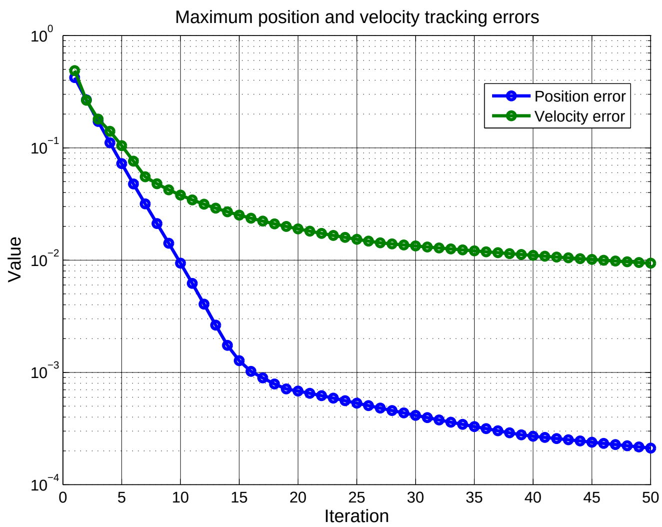

Figure 6.8: Maximum tracking errors vs. iteration number.

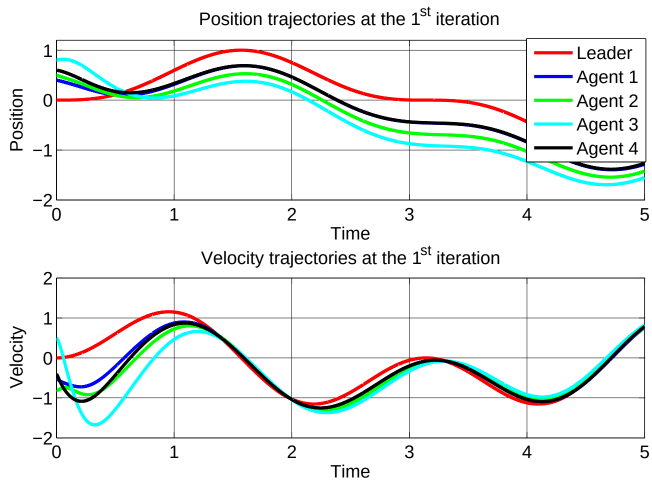

Figure 6.9: The trajectory profiles at the 1st iteration with initial rectifying action.

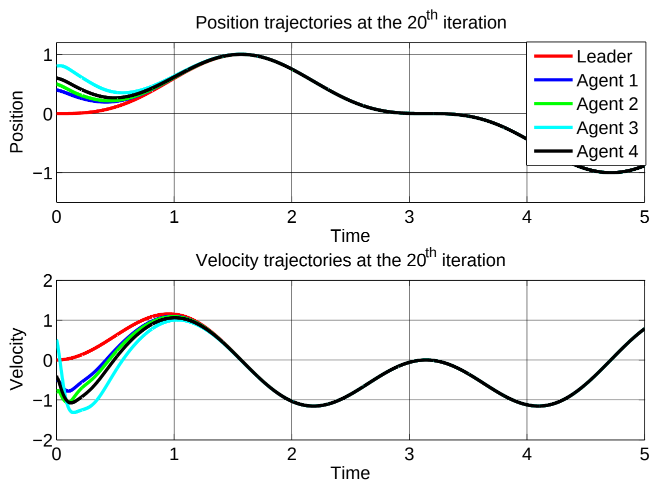

Figure 6.10: The trajectory profiles at the 20th iteration with initial rectifying action.

maximum velocity tracking error has been reduce to 1.93%.

To demonstrate the effectiveness of the initial rectifying action, let Assumption 6.5 hold, and the initial states of agents are chosen as $\mathbf{x}_1(0) = [0.4, -0.5]^T$, $\mathbf{x}_2(0) = [0.5, -0.8]^T$, $\mathbf{x}_3(0) = [0.8, 0.5]^T$, and $\mathbf{x}_4(0) = [0.6, -0.4]^T$. Obviously the initial states are not at the desired states. Matrix K is chosen such that

$$\bar{A} = \begin{bmatrix} 0 & 1 \\ -25 & -10 \end{bmatrix},$$

which has two eigenvalues located at -5 in the complex plane. By using the controllers (6.27) and (6.28), Figures. 6.9 and 6.10 depict the trajectory profiles at the 1st and 20th iterations respectively. It can be seen from Figure. 6.10 that the learning controllers with initial rectifying action demonstrate satisfactory performance even under the imperfect initial conditions.

6.6 Conclusion

Adaptive ILC algorithms are developed for a synchronization problem, which is formulated for a group of heterogeneous agents. Their dynamics are in general nonlinear form that do not satisfy the global Lipschitz condition. Although this group of agents are connected through communications, the desired trajectory is only known to a few of agents in the systems. By incorporating the desired trajectories as a part of the parametric uncertainty, the proposed ILC algorithm combines the parameter learning with robust control to handle both parametric uncertainties and lumped uncertainties. Meanwhile, the composite energy function plays an important role to ensure that the proposed ILC algorithm can synchronize all agents' trajectories to the desired one. Extensive synchronization examples verify the correctness of the developed methods.

Chapter 7

Synchronization for Networked Lagrangian Systems under Directed Graph

7.1 Background

Lagrangian system is an important class of systems, which can be used to model robotic manipulators, ground and underwater vehicles, helicopters, and satellites. Synchronization of networked Lagrangian systems has been reported in a number of publications. Leaderless synchronization algorithms are presented in Hou et al. (2009a, 2010); Ren (2009); Min et al. (2011), in which the final positions are constant and final velocities are zero. In the leader-follower tracking scenario, the final velocity is usually time-varying, which complicates the control problem. In Zhang et al. (2012), an adaptive backstepping-based method is developed for followers to track a dynamic leader. However, the velocity and acceleration signals of the leader have to be available to all followers. With the similar information assumption, a decentralized adaptive leader-follower control for multi-manipulator is proposed in Cheng et al. (2008). In Mei et al. (2011), effective controllers are developed for two cases, namely, the leader with constant velocity, and time-varying velocity. All the above mentioned works assume undirected graph for communication among followers. When the system parameters are precisely known, by applying the terminal sliding mode technique, a finite-time consensus tracking algorithm is developed in Khoo et al. (2009) under directed graph. The similar tracking error definition in Khoo et al. (2009) is adopted in Chen and Lewis (2011) to synchronize a group of uncertain robot manipulators, and the universal approximation ability of neural network is adopted to compensate for the model uncertainties. Similarly, the neural network is adopted in Cheng et al. (2010) to track a dynamic leader, and robust term is also included to counteract the external disturbance and approximation error. Furthermore, to avoid implementation dead loop, the acyclic communication is explicitly discussed in Cheng et al. (2010).

In this chapter, we consider the multi-agent synchronization problem by iterative learning control (ILC). Note that some robotic manipulator tracking algorithms are reported in the ILC literature (Tayebi, 2004; Sun et al., 2006; Ouyang et al., 2006). The problem formulation can be regarded as a single-leader-single-follower problem, where the information of the leader is known to the single follower. However, in the general multi-agent systems setup, the leader's information is usually only available to a small portion of the followers. It is currently not clear how the results in Tayebi (2004); Sun et al. (2006); Ouyang et al. (2006) can be generalized to the networked synchronization problem. In Chapter 6, an ILC rule for synchronization task is developed for parametric systems. However, the results rely on symmetric graph Laplacian but do not apply to directed graph. The main contributions of this chapter are summarized below. By fully utilizing the properties of Lagrangian systems, such as positive definite inertial matrix, skew symmetric and linear in parameter properties, a distributed ILC rule is constructed to achieve the synchronization task for networked Lagrangian systems. The plant parameters are assumed to be unknown, and the systems dynamics are subject to bounded repeatable disturbances. The developed control rule contains three components, namely, one proportional-plus-derivative (PD) term and two learning terms. The PD term drives the tracking error to zero, one learning term compensates for the model uncertainties, and the other learning term is used for rejecting the unknown disturbance. In addition, the communication graph is directed and acyclic, which reduces the communication burden compared to the undirected graph.

The rest of this chapter is organized as follows. In Section 7.2, the problem formulation and some useful facts are presented. The learning controller design and performance analysis are conducted in Section 7.3 under the identical initialization condition. In Section 7.4, the results are generalized to the alignment condition, which is more practical. To demonstrate the efficacy of the proposed learning controller, a numerical example is presented in Section 7.5. Finally, conclusions are drawn in Section 7.6.

7.2 Problem Description

The dynamics of the followers are described by the Lagrangian systems below

$$M_{i,j}(\mathbf{q}_{i,j}(t))\ddot{\mathbf{q}}_{i,j}(t) + C_{i,j}(\mathbf{q}_{i,j}(t), \dot{\mathbf{q}}_{i,j}(t))\dot{\mathbf{q}}_{i,j}(t) + G_{i,j}(\mathbf{q}_{i,j}(t)) = \tau_{i,j}(t) + \mathbf{w}_j(t), \quad (7.1)$$

where $j \in \mathscr{V}$ denotes the agent index, i represents the iteration index, t is the time argument and omitted in the following context for convenience, $\mathbf{q}_{i,j} \in \mathbb{R}^p$ is the vector of generalized coordinates, $M_{i,j}(\mathbf{q}_{i,j}) \in \mathbb{R}^{p\times p}$ is the inertial matrix, $C_{i,j}(\mathbf{q}_{i,j}, \dot{\mathbf{q}}_{i,j})\dot{\mathbf{q}}_{i,j} \in \mathbb{R}^p$

is the vector of Coriolis and centrifugal forces, $G_{i,j}(\mathbf{q}_{i,j})$ is the gravitational force, $\tau_{i,j}$ is the control input for the jth agent, and $\mathbf{w}_j$ is the bounded and repeatable disturbance.

The system represented by (1) is very general, and can be used to model a large class of systems. The disturbance $\mathbf{w}_j$ is assumed to be repeatable. This is a reasonable assumption due to the fact that the disturbances are repeatable in many practical systems. For instances, the parasitic voltage ripple of power supply systems, disturbances and frictions in rotary systems, atmospheric drag and solar radiation on the low earth orbit satellites. When the disturbance term is not repeatable, sliding mode technique can be applied to reject it if it is bounded (Cheng et al., 2010). As the focus of this work is on the learning control perspective, the non-repeatable disturbance is not considered.

According to Spong et al. (2006), the system modeled in (7.1) has three interesting and useful properties:

1. Positive definiteness: $M_{i,j}(\mathbf{q}_{i,j})$ is uniformly positive definite for any $\mathbf{q}_{i,j}$, i.e., there exist two positive constants α_j and β_j such that $0 < \alpha_j I \leq M_{i,j}(\mathbf{q}_{i,j}) \leq \beta_j I$.

2. Skew symmetric property: $\dot{M}_{i,j}(\mathbf{q}_{i,j}) - 2C_{i,j}(\mathbf{q}_{i,j}, \dot{\mathbf{q}}_{i,j})$ is skew symmetric.

3. Linear in parameter:

$$M_{i,j}(\mathbf{q}_{i,j})\mathbf{x} + C_{i,j}(\mathbf{q}_{i,j}, \dot{\mathbf{q}}_{i,j})\mathbf{y} + zG_{i,j}(\mathbf{q}_{i,j}) = Y_{i,j}(\mathbf{q}_{i,j}, \dot{\mathbf{q}}_{i,j}, \mathbf{x}, \mathbf{y}, z)\Theta_j,$$

   where $\mathbf{x}$, $\mathbf{y} \in \mathbb{R}^p$, z is a scalar, the regressor $Y_{i,j} \in \mathbb{R}^{p\times m}$ is a known function, and $\Theta_j \in \mathbb{R}^m$ is an unknown constant which represents the parameters of the system.

The leader's trajectory is given by $\mathbf{q}_0 \in \mathscr{C}^2[0,T]$, and $\mathbf{q}_0(t)$ for $t \in (0,T]$ is only known to a few of the followers. The control task is to generate an appropriate $\tau_{i,j}$ such that perfect tracking is achieved, i.e., $\lim_{i\to\infty} \left\|\mathbf{q}_{i,j} - \mathbf{q}_0\right\| = 0$. In the multi-agent problem setup, the control input has to be constructed by the local information under the communication topology $\bar{\mathscr{G}}$, i.e., $\tau_{i,j} = \tau_{i,j}(\ddot{\mathbf{q}}_{i,k}, \dot{\mathbf{q}}_{i,k}, \mathbf{q}_{i,k}), k \in \mathscr{N}_j$.

To restrict our discussion, the following assumptions about communication requirement and initialization conditions are imposed. These assumptions are extensively discussed in the previous chapters. Hence, they are not elaborated further.

Assumption 7.1 *The communication graph $\bar{\mathscr{G}}$ is acyclic and contains a spanning tree with the leader being the root.*

Assumption 7.2 *The initial states of all followers are reset to the desired initial state after each iteration, i.e., $\mathbf{q}_{i,j}(0) = \mathbf{q}_0(0), \dot{\mathbf{q}}_{i,j}(0) = \dot{\mathbf{q}}_0(0)$.*

Assumption 7.3 *The initial state of a follower at the current iteration is the final state of the previous iteration, namely, $\mathbf{q}_{i,j}(0) = \mathbf{q}_{i-1,j}(T), \dot{\mathbf{q}}_{i,j}(0) = \dot{\mathbf{q}}_{i-1,j}(T)$.*

The following Lemma is a useful result in multi-agent coordination.

Lemma 7.1 *(Ren and Cao, 2011, pp.9-10) All eigenvalues of $L+B$ have positive real parts if and only if the graph $\bar{\mathscr{G}}$ contains a spanning tree with the leader being the root, where L is the Laplacian matrix of $\mathscr{G}$, $\mathscr{G}$ is the communication graph among followers, and $B = \mathrm{diag}(b_1, \ldots, b_N)$.*

Remark 7.1 *When the communication graph is undirected and connected, the graph Laplacian is symmetric positive semi-definite. This property is usually utilized to construct an appropriate Lyapunov function to facilitate controller design as what we have done in Chapter 6. However, the Laplacian is asymmetric for directed graph in general, which makes the controller design more challenging.*

7.3 Controller Design and Performance Analysis

Define the actual tracking error $\mathbf{e}_{i,j} = \mathbf{q}_0 - \mathbf{q}_{i,j}$. As not all followers can obtain the tracking error, $\mathbf{e}_{i,j}$ cannot be used in the controller design. Following the convention in Khoo et al. (2009), the extended tracking errors are defined as follows

$$\xi_{i,j} = \sum_{k \in \mathcal{N}_j} a_{j,k}(\mathbf{q}_{i,k} - \mathbf{q}_{i,j}) + b_j(\mathbf{q}_0 - \mathbf{q}_{i,j}), \tag{7.2}$$

$$\zeta_{i,j} = \sum_{k \in \mathcal{N}_j} a_{j,k}(\dot{\mathbf{q}}_{i,k} - \dot{\mathbf{q}}_{i,j}) + b_j(\dot{\mathbf{q}}_0 - \dot{\mathbf{q}}_{i,j}), \tag{7.3}$$

where $\xi_{i,j}$ is the extended position tracking error, and $\zeta_{i,j}$ is the extended velocity tracking error. In particular, $\dot{\xi}_{i,j} = \zeta_{i,j}$.

From (7.2) and (7.3), the extended tracking errors can be represented in the compact forms as,

$$\xi_i = ((L+B) \otimes I)\mathbf{e}_i, \tag{7.4}$$

$$\zeta_i = ((L+B) \otimes I)\dot{\mathbf{e}}_i, \tag{7.5}$$

where ξ_i, ζ_i, and $\mathbf{e}_i$ are column stack vectors of $\xi_{i,j}$, $\zeta_{i,j}$, and $\mathbf{e}_{i,j}$ for $j = 1, \ldots, N$.

Based on Assumption 7.1 and Lemma 1, $L+B$ is of full rank since all the eigenvalues lie on the right half complex plane. Therefore, the minimum singular value $\underline{\sigma}(L+B) \neq 0$. As a result, it is straightforward to show that

$$\|\mathbf{e}_i\| \leq \frac{\|\xi_i\|}{\underline{\sigma}(L+B)}, \text{ and } \|\dot{\mathbf{e}}_i\| \leq \frac{\|\zeta_i\|}{\underline{\sigma}(L+B)},$$

by using the sub-multiplicative property of matrix norm. Therefore, if ξ_i and ζ_i converge to zero, $\mathbf{e}_i$ and $\dot{\mathbf{e}}_i$ converge to zero as well. Now the major issue is to design the distributed learning controllers, which drive ξ_i and ζ_i to zero along the iteration axis. To facilitate the controller design, define the following auxiliary variable

$$\mathbf{s}_{i,j} = \zeta_{i,j} + \lambda \xi_{i,j}, \tag{7.6}$$

where λ is a positive constant. Equation (7.6) can be treated as a stable filter with input $\mathbf{s}_{i,j}$. If $\|\mathbf{s}_{i,j}\| = 0$, $\xi_{i,j}$ and $\zeta_{i,j}$ converge to zero exponentially. In addition to $\|\mathbf{s}_{i,j}\| = 0$, if $\xi_{i,j}(0) = 0$ and $\zeta_{i,j}(0) = 0$, then $\xi_{i,j}(t) = 0$ and $\zeta_{i,j}(t) = 0$ for all $t \in [0,T]$.

From the system dynamics (7.1) and (7.6), the dynamics of the auxiliary variable $\mathbf{s}_{i,j}$, derived using the linear in parameter property, is given by

$$\begin{aligned}
& M_{i,j}\dot{\mathbf{s}}_{i,j} + C_{i,j}\mathbf{s}_{i,j} \\
= \; & -(d_j + b_j)(\tau_{i,j} + \mathbf{w}_j) + M_{i,j}\mathbf{x}_{i,j} + C_{i,j}\mathbf{y}_{i,j} + z_j G_{i,j} \\
= \; & -(d_j + b_j)(\tau_{i,j} + \mathbf{w}_j) + Y_{i,j}\Theta_j \qquad (7.7)
\end{aligned}$$

where $\mathbf{x}_{i,j} = \sum_{k\in\mathcal{N}_j} a_{j,k}\ddot{\mathbf{q}}_{i,k} + b_j\ddot{\mathbf{q}}_0 + \lambda\zeta_{i,j}$, $\mathbf{y}_{i,j} = \sum_{k\in\mathcal{N}_j} a_{j,k}\dot{\mathbf{q}}_{i,k} + b_j\dot{\mathbf{q}}_0 + \lambda\xi_{i,j}$, and $z_j = d_j + b_j$. Note that both $\mathbf{x}_{i,j}$ and $\mathbf{y}_{i,j}$ are distributed measurements.

The proposed controllers are

$$\tau_{i,j} = \frac{1}{d_j + b_j}(K\mathbf{s}_{i,j} + Y_{i,j}\hat{\Theta}_{i,j}) - \hat{\mathbf{w}}_{i,j}, \qquad (7.8)$$

$$\dot{\hat{\Theta}}_{i,j} = \gamma Y_{i,j}^T\mathbf{s}_{i,j},\ \hat{\Theta}_{i,j}(0) = \hat{\Theta}_{i-1,j}(T),\ \hat{\Theta}_{0,j}(0) = 0, \qquad (7.9)$$

$$\hat{\mathbf{w}}_{i,j} = \hat{\mathbf{w}}_{i-1,j} - \eta\mathbf{s}_{i,j},\ \hat{\mathbf{w}}_{0,j} = 0, \qquad (7.10)$$

where K is a positive definite matrix, γ and η are positive learning gains, $\hat{\Theta}_{i,j}$ is the estimate of constant unknown Θ_j, and $\hat{\mathbf{w}}_{i,j}$ is the estimate of time-varying but iteration-invariant disturbance $\mathbf{w}_j$. The controller (7.8) consists of three components, where $K\mathbf{s}_{i,j}$ is the proportional-plus-derivative term, which is commonly used in robotics control (Spong et al., 2006). $Y_{i,j}\hat{\Theta}_{i,j}$ compensates for the uncertainties in the system model, and $\hat{\mathbf{w}}_{i,j}$ is used for rejecting the repeatable disturbance. $\hat{\Theta}_{i,j}$ is updated by the differential updating rule (7.9), and $\hat{\mathbf{w}}_{i,j}$ is updated by the point-wise updating rule (7.10). As the communication graph in Assumption 7.1 contains a spanning tree with the leader being the root, $d_j + b_j \neq 0$. Therefore, the controller (7.8) is well defined.

Remark 7.2 *The regressor $Y_{i,j}$ contains the acceleration signal. However, acceleration measurement is usually not available in most mechanical systems. On the other hand, the velocity signal can be easily measured by tacho meters. Numerical differentiation of velocity generates large amount of noise, and should not be used for acceleration estimation. By using filtered differentiation (Slotine and Li, 1991) or the high gain observer (singular perturbation) (Khalil, 2002), a reasonable estimation of acceleration signal can be obtained. Detailed examples can be found in Lee and Khalil (1997); Islam and Liu (2010).*

Remark 7.3 *The differential updating rule (7.9) is applied since the system parameters are assumed to be constant. However, it is likely that the parameters are time-varying due to payload variations, mechanical wear, and aging. Similar to (7.10), point-wise updating rule can be used to handle the time-varying parameters. Therefore, the point-wise updating rule for time-varying unknown parameters is not detailed in this work.*

Theorem 7.1 *Under Assumptions 7.1 and 7.2, the closed loop system consisting of (7.1) and controllers (7.8)–(7.10) can ensure that the actual tracking error $\mathbf{e}_{i,j}(t)$ converges to zero in the sense of $\mathscr{L}^2[0,T]$ norm for $j = 1,\ldots,N$ as iteration number i goes to infinity. Moreover, $\tau_{i,j} \in \mathscr{L}^2[0,T]$ for any $j = 1,\ldots,N$, $i \in \mathbb{N}_{\geq 0}$.*

Proof: Consider the composite energy function (CEF)

$$E_i(t) = \sum_{j\in\mathscr{V}} E_{i,j}(t), \tag{7.11}$$

where the individual energy function (EF) on agent j is defined as

$$E_{i,j}(t) = \underbrace{\frac{1}{2}\mathbf{s}_{i,j}^T M_{i,j}\mathbf{s}_{i,j} + \frac{1}{2\gamma}\tilde{\Theta}_{i,j}^T\tilde{\Theta}_{i,j}}_{V_{i,j}} + \underbrace{\frac{d_j+b_j}{2\eta}\int_0^t \tilde{\mathbf{w}}_{i,j}^T\tilde{\mathbf{w}}_{i,j}\,dr}_{U_{i,j}}, \tag{7.12}$$

where $\tilde{\Theta}_{i,j} = \Theta_j - \hat{\Theta}_{i,j}$ and $\tilde{\mathbf{w}}_{i,j} = \mathbf{w}_j - \hat{\mathbf{w}}_{i,j}$.

From the definition of E_i, it is easy to see $E_i \geq 0$. For clarity, the proof is divided into three parts. In Part A, it is shown $E_i(T)$ is non-increasing along the iteration axis. The convergence of tracking error is shown in Part B, and in Part C, the boundedness of control input energy is proven.

Part A: Calculate ΔE_i

Let $\Delta E_{i,j} = E_{i,j} - E_{i-1,j}$. As such, $\Delta E_i = \sum_{j \in \mathscr{V}} \Delta E_{i,j}$, and $\Delta E_{i,j}$ can be written as

$$\Delta E_{i,j} = V_{i,j} - V_{i-1,j} + U_{i,j} - U_{i-1,j}. \tag{7.13}$$

Together with the error dynamics (7.7), controller (7.8), and the skew symmetric property, $V_{i,j}$ can written as

$$\begin{aligned}
V_{i,j} &= \int_0^t \dot{V}_{i,j}\, dr + V_{i,j}(0) \\
&= \int_0^t \left\{ \frac{1}{2}\mathbf{s}_{i,j}^T \dot{M}_{i,j}\mathbf{s}_{i,j} + \mathbf{s}_{i,j}^T M_{i,j}\dot{\mathbf{s}}_{i,j} + \frac{1}{\gamma}\tilde{\Theta}_{i,j}^T \dot{\tilde{\Theta}}_{i,j} \right\} dr + \frac{1}{2\gamma}\tilde{\Theta}_{i,j}(0)^T \tilde{\Theta}_{i,j}(0) \\
&= \int_0^t \left\{ -\mathbf{s}_{i,j}^T (K\mathbf{s}_{i,j} - Y_{i,j}\tilde{\Theta}_{i,j} + (d_j + b_j)\tilde{\mathbf{w}}_{i,j}) + \frac{1}{\gamma}\tilde{\Theta}_{i,j}^T \dot{\tilde{\Theta}}_{i,j} \right\} dr + \frac{1}{2\gamma}\tilde{\Theta}_{i,j}^T(0)\tilde{\Theta}_{i,j}(0).
\end{aligned} \tag{7.14}$$

By using the following equality

$$(\mathbf{a} - \mathbf{b})^T(\mathbf{a} - \mathbf{b}) - (\mathbf{a} - \mathbf{c})^T(\mathbf{a} - \mathbf{c}) = (\mathbf{c} - \mathbf{b})^T(2\mathbf{a} - \mathbf{b} - \mathbf{c}),$$

the difference of $U_{i,j}$ between two consecutive iterations can be evaluated as

$$\begin{aligned}
\Delta U_{i,j} &= U_{i,j} - U_{i-1,j} \\
&= \frac{d_j + b_j}{2\eta}\int_0^t \left\{ \tilde{\mathbf{w}}_{i,j}^T \tilde{\mathbf{w}}_{i,j} - \tilde{\mathbf{w}}_{i-1,j}^T \tilde{\mathbf{w}}_{i-1,j} \right\} dr \\
&= \frac{d_j + b_j}{2\eta}\int_0^t (\hat{\mathbf{w}}_{i-1,j} - \hat{\mathbf{w}}_{i,j})^T (2\mathbf{w}_j - \hat{\mathbf{w}}_{i,j} - \hat{\mathbf{w}}_{i-1,j}) dr \\
&= \frac{d_j + b_j}{2\eta}\int_0^t (\hat{\mathbf{w}}_{i-1,j} - \hat{\mathbf{w}}_{i,j})^T (2\mathbf{w}_j - 2\hat{\mathbf{w}}_{i,j} + \hat{\mathbf{w}}_{i,j} - \hat{\mathbf{w}}_{i-1,j}) dr \\
&= \frac{d_j + b_j}{2\eta}\int_0^t (\hat{\mathbf{w}}_{i-1,j} - \hat{\mathbf{w}}_{i,j})^T (2\tilde{\mathbf{w}}_{i,j} + \hat{\mathbf{w}}_{i,j} - \hat{\mathbf{w}}_{i-1,j}) dr \\
&\leq \frac{d_j + b_j}{\eta}\int_0^t (\hat{\mathbf{w}}_{i-1,j} - \hat{\mathbf{w}}_{i,j})^T \tilde{\mathbf{w}}_{i,j}\, dr.
\end{aligned} \tag{7.15}$$

Using equations (7.9), (7.10), (7.13), (7.14), and (7.15), $\Delta E_i(T)$ becomes

$$\begin{aligned}\Delta E_i(T) &= \sum_{j\in\mathscr{V}} \Big\{ -\int_0^T \mathbf{s}_{i,j}^T K \mathbf{s}_{i,j}\, dr \\ &\quad + \frac{1}{2\gamma}\tilde{\Theta}_{i,j}^T(0)\tilde{\Theta}_{i,j}(0) - \frac{1}{2\gamma}\tilde{\Theta}_{i-1,j}^T(T)\tilde{\Theta}_{i-1,j}(T) - \frac{1}{2}\mathbf{s}_{i-1,j}(T)M_{i-1,j}\mathbf{s}_{i-1,j}(T)\Big\} \\ &\leq -\sum_{j\in\mathscr{V}} \int_0^T \mathbf{s}_{i,j}^T K \mathbf{s}_{i,j}\, dr \leq 0. \end{aligned} \tag{7.16}$$

Therefore, $E_i(T)$ is non-increasing along the iteration axis.

Part B: Convergence of tracking error

Taking derivative of $E_{1,j}$ and substituting in the controllers (7.8)–(7.10) yield

$$\begin{aligned}\dot{E}_{1,j} &= -\mathbf{s}_{1,j}^T(K\mathbf{s}_{1,j} - Y_{1,j}\tilde{\Theta}_{1,j} + (d_j+b_j)\tilde{\mathbf{w}}_{1,j}) + \frac{1}{\gamma}\tilde{\Theta}_{1,j}^T\dot{\tilde{\Theta}}_{1,j} + \frac{d_j+b_j}{2\eta}\tilde{\mathbf{w}}_{1,j}^T\tilde{\mathbf{w}}_{1,j} \\ &= -\mathbf{s}_{1,j}^T K\mathbf{s}_{1,j} + \mathbf{s}_{1,j}^T Y_{1,j}\tilde{\Theta}_{1,j} + \frac{1}{\gamma}\tilde{\Theta}_{1,j}^T\dot{\tilde{\Theta}}_{1,j} - (d_j+b_j)\mathbf{s}_{1,j}^T\tilde{\mathbf{w}}_{1,j} + \frac{d_j+b_j}{2\eta}\tilde{\mathbf{w}}_{1,j}^T\tilde{\mathbf{w}}_{1,j} \\ &\leq -(d_j+b_j)\mathbf{s}_{1,j}^T(\mathbf{w}_j + \eta\mathbf{s}_{1,j}) + \frac{d_j+b_j}{2\eta}(\mathbf{w}_j+\eta\mathbf{s}_{1,j})^T(\mathbf{w}_j+\eta\mathbf{s}_{1,j}) \\ &\leq \frac{d_j+b_j}{2\eta}\mathbf{w}_j^T\mathbf{w}_j. \end{aligned} \tag{7.17}$$

As $\mathbf{w}_j$ is bounded, the boundedness of $\dot{E}_{1,j}$ can be concluded from (7.17). Furthermore, $E_{1,j}(t)$ can be calculated by integrating $\dot{E}_{1,j}$ over $[0,t]$, where $t \leq T$. Therefore, the boundedness of $E_{1,j}(t)$ is ensured for $t \in [0,T]$. $E_i(T)$ can be written as

$$\begin{aligned} E_i(T) &= E_1(T) + \sum_{k=2}^{i}\sum_{j\in\mathscr{V}} \Delta E_{k,j} \\ &= E_1(T) - \sum_{k=2}^{i}\sum_{j\in\mathscr{V}} \int_0^T \mathbf{s}_{k,j}^T K\mathbf{s}_{k,j}\, dr. \end{aligned} \tag{7.18}$$

The finiteness of $E_1(T)$ and positiveness of $E_i(T)$ leads to $\lim_{i\to\infty}\int_0^T \mathbf{s}_{i,j}^T\mathbf{s}_{i,j}dr = 0$. Therefore, $\mathbf{e}_{i,j}$ converges to zero in the sense of $\mathscr{L}^2[0,T]$ norm.

Part C: Boundedness property

Equation (7.18) implies that $E_i(T)$ is finite. As such, $E_{i,j}(T)$ is bounded for all agents, and there exist two finite constants P_j^1 and P_j^2 such that

$$\begin{aligned}
&\frac{1}{2\gamma}\tilde{\Theta}_{i-1,j}^T(0)\tilde{\Theta}_{i-1,j}(0) = \frac{1}{2\gamma}\tilde{\Theta}_{i,j}^T(T)\tilde{\Theta}_{i,j}(T) \leq P_j^1, \\
&\frac{d_j+b_j}{2\eta}\int_0^t \tilde{\mathbf{w}}_{i,j}^T\tilde{\mathbf{w}}_{i,j}\,dr \leq \frac{d_j+b_j}{2\eta}\int_0^T \tilde{\mathbf{w}}_{i,j}^T\tilde{\mathbf{w}}_{i,j}\,dr \leq P_j^2.
\end{aligned}$$

From Part A, it can be shown that

$$\begin{aligned}
\Delta E_{i,j}(t) \quad \leq \quad & \frac{1}{2\gamma}\tilde{\Theta}_{i,j}^T(0)\tilde{\Theta}_{i,j}(0) - \frac{1}{2\gamma}\tilde{\Theta}_{i-1,j}^T(t)\tilde{\Theta}_{i-1,j}(t) \\
& -\frac{1}{2}\mathbf{s}_{i-1,j}M_{i-1,j}\mathbf{s}_{i-1,j}.
\end{aligned}$$

Therefore, $E_{i,j}(t)$ can be expressed as

$$\begin{aligned}
E_{i,j}(t) \quad = \quad & E_{i-1,j}(t) + \Delta E_{i,j} \\
\leq \quad & \frac{1}{2\gamma}\tilde{\Theta}_{i,j}^T(0)\tilde{\Theta}_{i,j}(0) - \frac{1}{2\gamma}\tilde{\Theta}_{i-1,j}^T(t)\tilde{\Theta}_{i-1,j}(t) - \frac{1}{2}\mathbf{s}_{i-1,j}M_{i-1,j}\mathbf{s}_{i-1,j} + \frac{1}{2}\mathbf{s}_{i-1,j}^T M_{i-1,j}\mathbf{s}_{i-1,j} \\
& + \frac{1}{2\gamma}\tilde{\Theta}_{i-1,j}^T(t)\tilde{\Theta}_{i-1,j}(t) + \frac{d_j+b_j}{2\eta}\int_0^t \tilde{\mathbf{w}}_{i-1,j}^T\tilde{\mathbf{w}}_{i-1,j}\,dr \\
\leq \quad & \frac{1}{2\gamma}\tilde{\Theta}_{i,j}^T(0)\tilde{\Theta}_{i,j}(0) + \frac{d_j+b_j}{2\eta}\int_0^t \tilde{\mathbf{w}}_{i-1,j}^T\tilde{\mathbf{w}}_{i-1,j}\,dr \\
\leq \quad & P_j^1 + P_j^2.
\end{aligned}$$

Hence, $E_{i,j}(t)$ is bounded for all $t \in [0,T]$. From the definition of $E_{i,j}(t)$ in (7.12), it can be concluded that $\mathbf{s}_{i,j}$ and $\hat{\Theta}_{i,j}$ are bounded, and $\hat{\mathbf{w}}_j \in \mathscr{L}^2[0,T]$. Therefore, $\tau_{i,j} \in \mathscr{L}^2[0,T]$. ■

Remark 7.4 *In today's control systems, most controllers are realized by digital computers. In the controller design, the regressor $Y_{i,j}$ requires the acceleration signals of agents in the neighborhood $\mathscr{N}_j$ at the current time instance, which is not measurable. Therefore, the acceleration signals have to be estimated from velocity measurements. For example, pass the measured velocity through the filter $\frac{\alpha p}{p+\alpha}$, where p is the Laplace variable, and $\alpha \gg 1$ is a design parameter. Applying the zero-order hold operation, the* *discrete implementation of the differentiation is (Slotine and Li, 1991, pp. 202)*

$$\begin{aligned} \nu_{i,j}((k+1)h) &= a_1\nu_{i,j}(kh) + a_2\dot{\mathbf{q}}_{i,j}(kh), \\ \psi((k+1)h) &= \alpha(\dot{\mathbf{q}}_{i,j}(kh) - \nu_{i,j}((k+1)h)), \end{aligned}$$

where k is the discrete-time index, h is the sampling time, $\nu_{i,j}$ is the internal state, $\psi_{i,j}$ is the estimation of $\ddot{\mathbf{q}}_{i,j}$, $a_1 = e^{-\alpha h}$, and $a_2 = 1 - a_1$.

If the communication graph contains cycles, the acceleration signal may form a closed loop dynamics, which is likely to be unstable. This observation can be verified by numerical studies. However, stability analysis of such a sampled-data system is extremely difficult due to the nonlinear terms in the closed loop systems. To rule out such kind of situations, the directed acyclic graph is required for communication.

Remark 7.5 *In case the communication graph contains cycles, some of the edges have to be removed in order to make the graph acyclic. Alternatively, an agent simply does not use the data received from certain channels so that Assumption 7.1 is fulfilled.*

If the upper and lower bounds of $\mathbf{w}_j$ are known, the projection operator can be used to achieve a stronger convergence result. The projection operator can be defined as,

$$\mathrm{proj}(z) = \begin{cases} \underline{z}^* & \text{if } z < \underline{z}^* \\ z & \underline{z}^* \leq z \leq \overline{z}^* \\ \overline{z}^* & z > \overline{z}^* \end{cases}$$

where $\overline{z}^*$ and $\underline{z}^*$ are the upper and lower bounds of z. When the argument is a vector, the projection operator is defined element-wise. By applying the projection operator to (7.10), the controller becomes

$$\hat{\mathbf{w}}_{i,j} = \mathrm{proj}(\hat{\mathbf{w}}_{i-1,j}) - \eta\mathbf{s}_{i,j},\ \hat{\mathbf{w}}_{0,j} = 0. \tag{7.19}$$

Corollary 7.1 *Under Assumptions 7.1 and 7.2, the closed loop system consisting of (7.1) and controllers (7.8), (7.10), (7.19) can ensure that the actual tracking error* $\mathbf{e}_{i,j}(t)$ *converges to zero uniformly for* $j = 1, \ldots, N$ *as iteration number i goes to infinity. Moreover,* $\tau_{i,j}$ *is bounded for any* $j = 1, \ldots, N$*,* $i \in \mathbb{N}_{\geq 0}$*.*

Proof: Following the similar procedure as used in the proof of Theorem 7.1, it can be shown that $\mathbf{s}_{i,j}$ and $\hat{\Theta}_{i,j}$ are bounded. Therefore, the boundedness of $\hat{\mathbf{w}}_{i,j}$ can be concluded from (7.19). The controller (7.8) leads to the boundedness of control input $\tau_{i,j}$. As such $\dot{\mathbf{s}}_{i,j}$ is finite. Together with (7.18), the uniform convergence of $\mathbf{s}_{i,j}$ and $\mathbf{e}_{i,j}$ can be concluded. ■

Remark 7.6 *The projection operation in (7.19) requires priori information about the disturbance. When the upper and lower bounds of the disturbances are unknown, the upper and lower bounds can be chosen as arbitrarily large constants.*

7.4 Extension to Alignment Condition

In addition to the Assumption 7.3, the desired trajectory has to be closed, i.e., $\mathbf{q}_0(0) = \mathbf{q}_0(T)$ and $\dot{\mathbf{q}}_0(0) = \dot{\mathbf{q}}_0(T)$. The results in Section 7.3 are still valid when the alignment assumption is assumed. Although this assumption seems to be more stringent than the *i.i.c.*, it is more practical. For example, for a satellite orbiting the earth periodically, the position and speed at the beginning of the current cycle are identical to the ones at terminal time of the previous cycle.

Theorem 7.2 *Under Assumptions 7.1 and 7.3, if the leader's trajectory is closed, the closed loop system consisting of (7.1) and controllers (7.8)–(7.10) can ensure that the actual tracking error* $\mathbf{e}_{i,j}(t)$ *converges to zero in the sense of* $\mathscr{L}^2[0,T]$ *norm for* $j =$

$1,\ldots,N$ as iteration number i goes to infinity. Moreover, $\tau_{i,j} \in \mathscr{L}^2[0,T]$ for any $j = 1,\ldots,N$, $i \in \mathbb{N}_{\geq 0}$.

Proof: Consider the same CEF in (7.11). Since the *i.i.c.* is not assumed, $\mathbf{s}_{i,j}^T(0)M_{i,j}\mathbf{s}_{i,j}(0) \neq 0$. Similar to (7.14), $V_{i,j}$ can be expressed as

$$\begin{aligned} V_{i,j} &= \int_0^t \left\{ -\mathbf{s}_{i,j}^T(K\mathbf{s}_{i,j} - Y_{i,j}\tilde{\Theta}_{i,j} + (d_j + b_j)\tilde{\mathbf{w}}_{i,j}) + \frac{1}{\gamma}\tilde{\Theta}_{i,j}^T\dot{\tilde{\Theta}}_{i,j} \right\} dr + \frac{1}{2\gamma}\tilde{\Theta}_{i,j}^T(0)\tilde{\Theta}_{i,j}(0) \\ &\quad + \frac{1}{2}\mathbf{s}_{i,j}^T(0)M_{i,j}\mathbf{s}_{i,j}(0). \end{aligned} \tag{7.20}$$

Assumption 7.3 implies that

$$\mathbf{s}_{i,j}^T(0)M_{i,j}\mathbf{s}_{i,j}(0) = \mathbf{s}_{i-1,j}^T(T)M_{i-1,j}\mathbf{s}_{i-1,j}(T).$$

Together with (7.16) and (7.20), $\Delta E_i(T)$ can be derived as

$$\Delta E_i(T) = -\sum_{j\in\mathscr{V}} \int_0^T \mathbf{s}_{i,j}^T K\mathbf{s}_{i,j}\, dr \leq 0.$$

Following the steps in the proof to Theorem 7.1, Theorem 7.2 can be concluded. ■

Analogous to Corollary 7.1, the projection operation can also be applied under Assumption 7.3. As such, we have the corresponding Corollary 7.2.

Corollary 7.2 *Under Assumptions 7.1 and 7.3, if the leader's trajectory is closed, the closed loop system consisting of (7.1) and controllers (7.8), (7.10), (7.19) can ensure that the actual tracking error $\mathbf{e}_{i,j}(t)$ converges to zero uniformly for $j = 1,\ldots,N$ as iteration number i goes to infinity. Moreover, $\tau_{i,j}$ is bounded for any $j = 1,\ldots,N$, $i \in \mathbb{N}_{\geq 0}$.*

7.5 Illustrative Example

To demonstrate the effectiveness of the developed methods, consider four networked two-link robotic arms, which are modeled by the example in (Slotine and Li, 1991, pp.

396) using

$$M_{i,j}(\mathbf{q}_{i,j})\ddot{\mathbf{q}}_{i,j} + C_{i,j}(\mathbf{q}_{i,j}, \dot{\mathbf{q}}_{i,j})\dot{\mathbf{q}}_{i,j} = \tau_{i,j} + \mathbf{w}_j.$$

The detailed definitions of $M_{i,j}$ and $C_{i,j}$ are given in Slotine and Li (1991). The nominal values of the plant parameters are $m_1^0 = 1\,kg$, $l_1^0 = 1\,m$, $m_e^0 = 2\,kg$, $\delta_e = 30^o$, $I_1^0 = 0.12\,kg \cdot m^2$, $l_{c1}^0 = 0.5\,m$, $I_e^0 = 0.25\,kg \cdot m^2$, and $l_{ce}^0 = 0.6\,m$. To demonstrate the capability of the proposed method in dealing with heterogeneous agent systems, the actual parameters for agent j are set as follows: $\omega^j = (1+0.1j)\omega$, where $\omega \in \{m_1^0, l_1^0, m_e^0, I_1^0, l_{c1}^0, I_e^0, l_{ce}^0\}$. $\mathbf{w}_j$ can be used to model the iteration-invariant input disturbance. For instance, the frictions in rotary motion systems. In this example, the input disturbance is set as $\mathbf{w}_j = \mathbf{1}\sin(j \cdot t)$. It is worth noting that all the parameters and disturbances are assumed to be unknown in the actual simulation. Therefore, the system parameters such as m_1^j, l_1^j, m_e^j, etc. are lumped into Θ_j, and estimated by $\hat{\Theta}_{i,j}$. Similarly, the unknown disturbance is estimated by $\hat{\mathbf{w}}_{i,j}$.

The leader's trajectory is a closed curve, and $\mathbf{q}_0 = [\sin(t), \cos(t)]^T$ for $t \in [0, 2\pi]$. The communication among the followers and the leader is described by a directed acyclic graph in Figure. 7.1. Only the first follower has access to the leader's trajectory, and all other followers have to learn the desired trajectory from their neighbors. The alignment condition is assumed for the numerical study. The initial states of the followers are $\mathbf{q}_{1,1} = [1, 0.1]^T$, $\mathbf{q}_{1,2} = [2, 2]^T$, $\mathbf{q}_{1,3} = [0.5, 0.3]^T$, $\mathbf{q}_{1,4} = [0.8, 0.4]^T$, $\dot{\mathbf{q}}_{1,1} = [0.2, 0.6]^T$, $\dot{\mathbf{q}}_{1,2} = [0.4, 0.7]^T$, $\dot{\mathbf{q}}_{1,3} = [0.6, 0.3]^T$, and $\dot{\mathbf{q}}_{1,4} = [0.5, 0.5]^T$.

Controllers (7.8)–(7.10) are applied to the networked systems. There are four controller parameters to tune, namely, K, γ, η, and λ. Based on the theorems and corollaries in this work, if K is positive definite and all the rest parameters are positive, individual tracking error asymptotically converges to zero in the sense of $\mathscr{L}^2[0,T]$ norm.

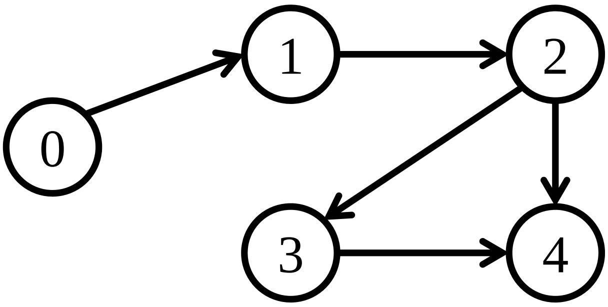

Figure 7.1: Directed acyclic graph for describing the communication among agents.

However, the controller parameters directly affect the transient performance. Generally speaking, when the parameters are chosen with small values, the control profiles are smooth and the convergence rate is slow. Whereas, if the parameters have large magnitudes, the convergence rate is very fast. However, when the values go beyond certain limits, the control profiles become very oscillatory due to the fast learning in estimation of unknown terms. In the numerical study, the controller parameters are selected as $K = \text{diag}(7, 14)$, $\gamma = 1$, $\eta = 1$, and $\lambda = 1$.

The trajectory profiles of all agents at the 1st iteration are shown in Figure. 7.2, where it can be seen that the followers' trajectories have large deviations from the leader's. The developed controllers enable the followers to learn and improve the tracking performances from iteration to iteration. In contrast to Figure. 7.2, the trajectory profiles at the 70th iteration are shown in Figure. 7.3, where it can be seen that all the trajectories are indistinguishable. Define the maximum position errors at the ith iteration as $\max_{j\in\mathscr{V}} \|[\mathbf{e}_{i,j}]_1\|$ for the first component, and $\max_{j\in\mathscr{V}} \|[\mathbf{e}_{i,j}]_2\|$ for the second one. The maximum tracking error along the iteration profile is plotted in Figure. 7.4. The maximum position errors of the first and second generalized coordinates at the 70th iteration have been respectively reduced to 0.67% and 0.72% of the ones at the 1st iteration. The initial control inputs at the 1st iteration are described in Figure. 7.5. For each robotic arm, the input should reject the external disturbance and compensate for the model uncertainties. As the learning process evolves along the iteration domain, the control inputs gradually converges to the ones depicted in Figure. 7.6, in which the trajectories are smooth.

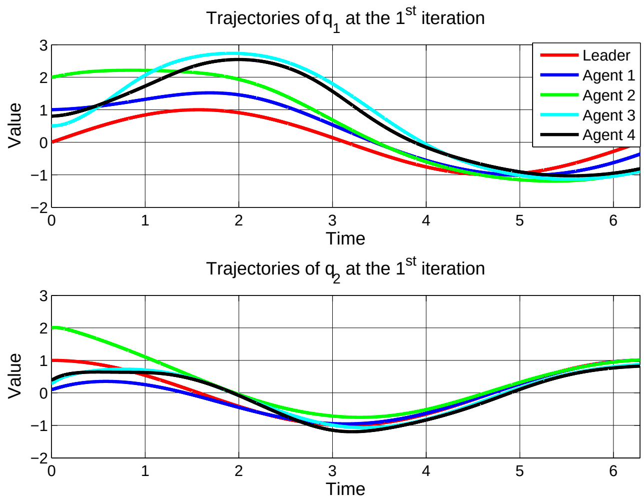

Figure 7.2: Trajectory profiles at the 1st iteration.

7.6 Conclusion

A leader-follower synchronization problem is formulated in the ILC framework for a group of Lagrangian systems with model uncertainties and repeatable external disturbances. By fully utilizing the properties of Lagrangian systems, learning controllers for the synchronization problem are developed under the identical initialization condition and alignment condition respectively. It has been shown that the learning rules can effectively deal with both constant and time-varying unknowns. In contrast to many ILC works for synchronization problem that take advantages of symmetric Laplacian for undirected graph, it turns out that the directed acyclic graph is sufficient for communication among agents. Numerical study supports the theoretical results.

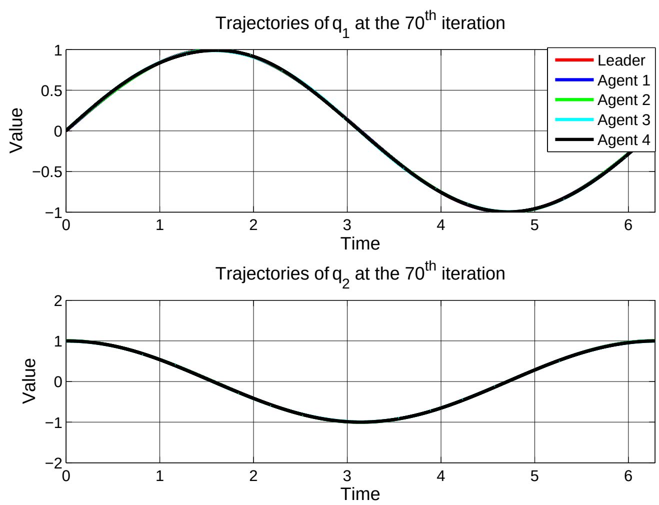

Figure 7.3: Trajectory profiles at the 70th iteration, all trajectories overlap with each other.

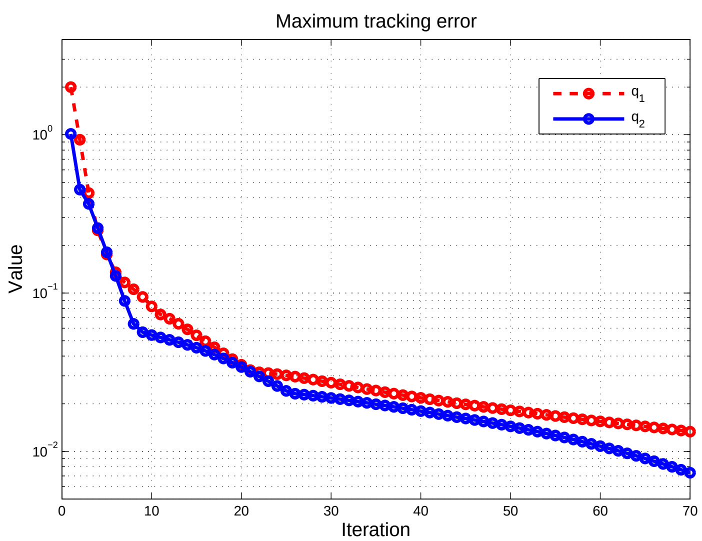

Figure 7.4: Maximum tracking error profile.

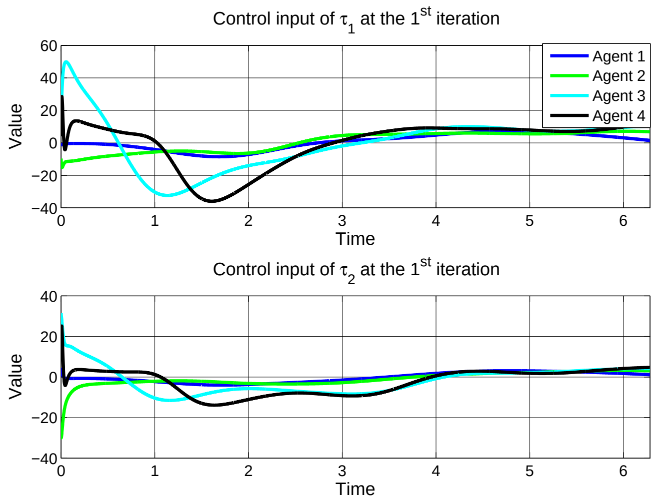

Figure 7.5: Control input profiles at the 1st iteration.

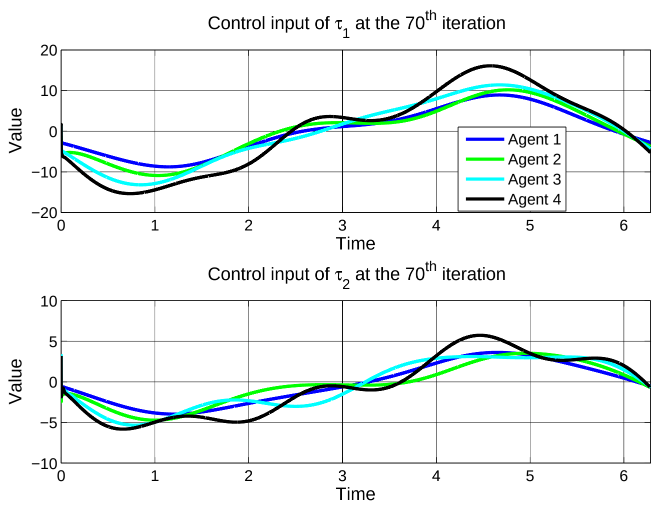

Figure 7.6: Control input profiles at the 70th iteration.

Chapter 8

Conclusion and Future Work

8.1 Conclusion

This thesis studies the multi-agent coordination and control problem from iterative learning control (ILC) perspective. As consensus tracking is one of the most important coordination problems in the multi-agent systems, and various coordination problems can be formulated and solved in the consensus framework, for example the formation control of multiple aerial vehicles, rendezvous of ground robots, sensor fusion, area coverage by multiple robots, and distributed optimizations. Therefore, consensus tracking is chosen as the main problem studied in this thesis.

We follow the two main approaches in ILC, namely contraction-mapping (CM) and composite energy function (CEF) based methods, to investigate the consensus tracking problems. First, the consensus tracking problem is formulated for a group of global Lipschitz continuous agents under the fixed communication assumption and perfect initialization condition. Distributed D-type iterative learning rules are developed for all of the followers to track a dynamic leader agent. Next, we relax the communication assumption by considering the intermittent interactions among agents. It is shown that the consensus tracking can be achieved under very mild condition that the communication graph is uniformly strongly connected along the iteration axis. Furthermore, to remove the stringent initialization condition, we only assume that the follower agent is reset to the same initial state at the beginning of every iteration which may not be the same as the leader's initial state. It is shown that the previous D-type learning rule is still convergent. However, the final tracking trajectories may have large deviations from the leader's. To improve the control performance under the imperfect initialization condition, a PD-type learning rule is developed for the multi-agent systems. The PD-type learning rule has two advantages. On the one hand, it can ensure learning convergence; on the other hand, it offers the controller designer additional freedom to tune the final tracking performance. The results summarized above are derived based on the CM method. Hence, they are only applicable to the linear and global Lipschitz nonlinear systems. CM method for local Lipschitz systems has litter progress for the past 30 years. The traditional λ-norm fails to construct a contraction-mapping for local Lipschitz systems as we are unable to get a reasonable estimate of the upper bound of system state. By combining the Lyapunov analysis and CM analysis methods, we can show that CM based ILC can be applied to several classes of local Lipschitz systems such as stable systems with quadratic Lyapunov functions, exponentially stable systems, systems with bounded drift terms, and uniformly bounded energy bounded state systems under control saturation. This part of the results greatly complements the existing ILC literature, and also implies that the learning rules developed for multi-agent systems can be applied a large scale of nonlinear systems. Finally, CEF based ILC is adopted to study local Lipschitz systems whose dynamics can be linearly parameterized and full state information can be used for feedback. The controller design requires undirected communication. Initial rectifying action control is employed to deal with the imperfect initialization condition. The controller is applicable to high-order systems and able to reject the state dependent disturbance. This part of the results requires symmetric Laplacian, namely the undirected communication graph. Finally, we develop a set of distributed learning rules to synchronize networked Lagrangian systems under directed acyclic graph. The learning rules fully utilize the properties of Lagrangian systems and CEF approach is adopted to analyze their convergence.

8.2 Future Work

This thesis develops the general design principle for multi-agent coordination by ILC approach. Some specific problems like switching communication, imperfect initialization condition, and local Lipschitz continuous systems are investigated. Although many interesting results are presented, there are many other issues remaining unsolved not only in the multi-agent set up but also in the ILC literature. Here we list some of the open problems.

1. To deal with imperfect initialization condition, PD-type learning rule is developed in Chapter 4. By using the similar idea and mathematical technique, it is possible to extend the PD-type learning rule to PID-type learning rule. The PID-type learning rule offers more freedom to tune the final tracking performance.

2. In the problem description, the leader's trajectory is assumed to be iteration-invariant. When the leader's trajectory changes from iteration to iteration, the existing learning rule does not work any more. Tracking an iteration-varying reference trajectory is still a challenging problem in the ILC literature. If the reference trajectories can be characterized by a high order internal model (HOIM) over

the iteration domain, the HOIM can be incorporated in the learning rule design, for example Liu et al. (2010). We can adopt the HOIM approach in the multi-agent coordination problem. However, all the followers are required to know the HOIM. Whereas, the HOIM is regarded as the global information. Such kind of approach conflicts with the spirit of distributed controller design because only a few of the followers can access the leader's information. The learning rule that does not require the HOIM is desired.

3. In Chapter 5, we show that the P-type learning rule can be applied to several local Lipschitz systems. One sufficient condition says if the unstable factor in the unforced systems satisfies the global Lipschitz condition, then P-type rule is applicable to the system. At this stage, we do not know whether the controller converges or not when the unstable factor only satisfies the local Lipschitz condition. Based on our numerical study, P-type rule may converge in certain cases. However, we are unable to identify a general result in this case. Besides D-type learning rule for local Lipschitz systems remains unknown and is worth investigation.

4. ILC is a kind of partial model free control method. However, it is a waste if we do not use the inherent system properties when they are known. For example the Lagrangian system has three useful properties, linear in parameter, positive definiteness, and skewed symmetric property. It is possible that we can take advantages of these features to improve control performance. In Chapter 7 the three properties are fully utilized in the learning rule design. Unlike many existing results which rely on symmetric interaction topology, the controller works under directed graph. However, the graph is required to be acyclic. Otherwise, it may

lead to instability problem in the learning rule. It is an open problem to design learning rule that works under the general directed graph by CEF approach. The main challenge is to find a suitable energy function which facilitates distributed learning rule design.

5. In Yang and Xu (2014) we devise an input sharing mechanism for a group of identical linear agent systems to perform leader-follower consensus tracking by CM approach. Traditional ILC rule has only one learning resource, that is the correction term. The input sharing mechanism allows the followers to share their learned information with their neighbors. It is likely that the additional learning resource will enhance the learning experience. In the paper, numerical examples demonstrate that the input sharing mechanism not only increase the convergence rate, but also smooth the transient performance. However, quantitative performance improvement has not yet been examined. In addition, the idea of input sharing is new and may lead to other novel controllers.

Bibliography

Hyo-Sung Ahn and Yangquan Chen. Iterative learning control for multi-agent formation. In *ICROS-SICE International Joint Conference*, pages 3111–3116, Fukuoka International Congress Center, Japan, 18-21 August 2009.

Hyo-Sung Ahn, Yangquan Chen, and Kevin L Moore. Iterative learning control: Brief survey and categorization. *IEEE Transactions on Systems, Man, and Cybernetics - Part C: Applications and Reviews*, 37(6):1099–1121, 2007.

Hyo-Sung Ahn, Kevin L Moore, and Yangquan Chen. Trajectory-keeping in satellite formation flying via robust periodic learning control. *International Journal of Robust and Nonlinear Control*, 20(14):1655–1666, 2010.

Claudio Altafini. Consensus problems on networks with antagonistic interactions. *IEEE Transactions on Automatic Control*, 58(4):935–946, 2013.

David Angeli, Eduardo D Sontag, and Yuan Wang. Further equivalences and semiglobal versions of integral input to state stability. *Dynamics and Control*, 10(2):127 – 149, 2000a.

David Angeli, Eduardo D Sontag, and Yuan Wang. A characterization of integral input to state stability. *IEEE Transactions on Automatic Control*, 45(6):1082–1097, 2000b.

Suguru Arimoto, Sadao Kawamura, and Fumio Miyazaki. Bettering operation of robots by learning. *Journal of Robotic Systems*, 1(2):123–140, 1984.

He Bai and John T Wen. Cooperative load transport: A formation-control perspective. *IEEE Transactions on Robotics*, 26(4):742–750, 2010.

Norman Biggs. *Algebraic Graph Theory*. Cambridge University Press, second edition, 1994.

D A Bristow, M Tharayil, and A G Alleyne. A survey of iterative learning control a learning-based method for high-performance tracking control. *IEEE Control Systems Magazine*, 26:96–114, 2006.

Ming Cao, A S Morse, and B D O Anderson. Coordination of an asynchronous multi-agent system via averaging. In *Proceedings of the 16th IFAC World Congress*, Czech Republic, 2005.

Yongcan Cao, Wenwu Yu, Wei Ren, and Guangrong Chen. An overview of recent progress in the study of distributed multi-agent coordination. *IEEE Transactions on Industrial Informatics*, 9(1):427–438, 2013.

Gang Chen and Frank L Lewis. Distributed adaptive tracking control for synchronization of unknown networked lagrangian systems. *IEEE Transactions on Systems, Man, and Cybernetics - Part B: Cybernetics*, 41(3):805–816, 2011.

Yangquan Chen, C Wen, Z Gong, and Mingxuan Sun. A robust high-order ptype iterative learning controller using current iteration tracking error. *International Journal of Control*, 68(2):331–342, 1997.

Yangquan Chen, C Wen, Z Gong, and Mingxuan Sun. An iterative learning controller with initial state learning. *IEEE Transaction on Automatic Control*, 44(2):371–375, 1999.

Long Cheng, Zeng-Guang Hou, and Min Tan. Decentralized adaptive leader-follower control of multi-manipulator system with uncertain dynamics. In *Proceedings of The 34th Annual Conference of The IEEE Industrial Electronics Society*, pages 1608–1613, November 2008.

Long Cheng, Zeng-Guang Hou, Min Tan, Yingzi Lin, and Wenjun Zhang. Neural-network-based adaptive leader-following control for multiagent systems with uncertainties. *IEEE Transactions on Neural Networks*, 21(8):1351–1358, 2010.

Ronghu Chi, Zhongsheng Hou, and Jian-Xin Xu. Adaptive ilc for a class of discrete-time systems with iteration-varying trajectory and random initial condition. *Automatica*, 44:2207–2213, 2008.

Tommy W S Chow and Yong Fang. An iterative learning control method for continuous-time systems based on 2-d system theory. *IEEE Transactions on Circuits and SystemsII: Fundamental Theory and Applications*, 45(4):683–689, 1998.

Jorge Cortex. Finite-time convergent gradient flows with applications to network consensus. *Automatica*, 42(11):1993–2000, 2006.

Zhi-Sheng Duan and Guan-Rong Chen. Does the eigenratio λ_2/λ_n represent the synchronizability of a complex network? *Chinese Physics B*, 21(8):080506, 2012.

Lei Fang and Panos J Antsaklis. On communication requirements for multi-agent consensus seeking. *Networked Embedded Sensing and Control, Proceedings of Workshop NESC05*, pages 53–68, 2006.

Yong Fang and Tommy W S Chow. 2-d analysis for iterative learning controller for discrete-time systems with variable initial conditions. *IEEE Transactions on Circuits and SystemsłI: Fundamental Theory and Applications*, 50(5):722–727, 2003.

Yuko Hatano and Mehran Mesbahi. Agreement over random networks. *IEEE Transactions on Automatic Control*, 50(11):1867–1872, 2005.

Yiguang Hong, Jiangping Hu, and Linxin Gao. Tracking control for multi-agent consensus with an active leader and variable topology. *Automatica*, 42(7):1177–1182, 2006.

Roger A Horn and Charles R Johnson. *Matrix Analysis*. Cambridge University Press, 1985.

Zeng-Guang Hou, Long Cheng, and Min Tan. Decentralized robust adaptive control for the multiagent system consensus problem using neural networks. *IEEE Transactions on Systems, Man, and Cybernetics - Part B: Cybernetics*, 39(3):636–647, 2009a.

Zeng-Guang Hou, Long Cheng, and Min Tan. Decentralized robust adaptive control for the multiagent system consensus problem using neural networks. *IEEE Transactions on Systems, Man, And CyberneticsłPart B: Cybernetics*, 39(3):636–647, 2009b.

Zeng-Guang Hou, Long Cheng, Min Tan, and XuWang. Distributed adaptive coordinated control of multi-manipulator systems using neural networks. In *Robot Intelligence: An Advanced Knowledge Processing Approach*, chapter 3, pages 44 – 69. Springer-Verlag, London, August 2010.

Jie Huang. Remarks on 'synchronized output regulation of linear networked systems'. *IEEE Transactions on Automatic Control*, 56(3):630–631, 2011.

S Islam and P X Liu. Adaptive iterative learning control for robot manipulators without using velocity signals. In *Proceedings of IEEE/ASME Iternational Conference on Advanced Intelligent Mechatronics*, pages 1293–1298, Montreal, Canada, 6-9 July 2010.

Ali Jadbabaie, Jie Lin, and A Stephen Morse. Coordination of groups of mobile autonomous agents using nearest neighbor rules. *IEEE Transactions on Automatic Control*, 48(6):988–1001, 2003.

Zhong-Ping Jiang and Yuan Wang. Input-to-state stability for discrete-time nonlinear systems. *Automatica*, 37(6):857–869, 2001.

Hassan K Khalil. *Nonlinear Systems*. Prentice Hall, third edition, 2002.

Suiyang Khoo, Lihua Xie, and Zhihong Man. Robust finite-time consensus tracking algorithm for multirobot systems. *IEEE/ASME Transactions on Mechatronics*, 14 (2):219–228, 2009.

Kang Woong Lee and Hassan K Khalil. Adaptive output feedback control of robot manipulators using high-gain observer. *International Journal of Control*, 67:869–886, 1997.

Jinsha Li and Junmin Li. Adaptive iterative learning control for coordination of second-order multi-agent systems. *International Journal of Robust and Nonlinear Control*, 2013. doi: 10.1002/rnc.3055. URL `http://dx.doi.org/10.1002/rnc.3055`.

Shihua Li, Haibo Du, and Xiangze Lin. Finite-time consensus algorithm for multi-agent systems with double-integrator dynamics. *Automatica*, 47:1706–1712, 2011.

Zhongkui Li, Zhisheng Duan, Guanrong Chen, and Lin Huang. Consensus of multiagent systems and synchronization of complex networks: A unified viewpoint. *IEEE Transactions on Circuits and Systems - I*, 57(1):213–224, 2010.

Chunping Liu, Jian-Xin Xu, and Jun Wu. On iterative learning control with high-order internal models. *International journal of Adaptive Control and Signal Processing*, 24 (9):731–742, 2010.

Yang Liu and Yingmin Jia. An iterative learning approach to formation control of multi-agent systems. *Systems & Control Letters*, 61(1):148–154, 2012.

Richard W Longman. Iterative learning control and repetitive control for engineering practice. *International Journal of Control*, 73(10):930–954, 2000.

Cui-Qin Ma and Ji-Feng Zhang. Necessary and sufficient conditions for consensusability of linear multi-agent systems. *IEEE Transactions on Automatic Control*, 55(5): 1263–1268, 2010.

Jie Mei, Wei Ren, and Guangfu Ma. Distributed coordinated tracking with a dynamic leader for multiple euler-lagrange systems. *IEEE Transactions on Automatic Control*, 56(6):1415–1421, 2011.

Deyuan Meng and Yingmin Jia. Iterative learning approaches to design finite-time consensus protocols for multi-agent systems. *Systems & Control Letters*, 61(1):187–194, 2012.

Deyuan Meng, Yingmin Jia, Junping Du, and Fashan Yu. Tracking control over a finite interval for multi-agent systems with a time-varying reference trajectory. *Systems & Control Letters*, 61(7):807–818, 2012.

Deyuan Meng, Yingmin Jia, and Junping Du. Multi-agent iterative learning control with communication topologies dynamically changing in two directions. *Control Theory & Applications, IET*, 7(2):260–271, 2013a.

Deyuan Meng, Yingmin Jia, Junping Du, Jun Zhang, and Wenlin Li. Formation learning algorithms for mobile agents subject to 2-d dynamically changing topologies. In *IEEE American Control Conference*, pages 5165–5170, Washington, DC, 17-19 June 2013b.

Deyuan Meng, Yingmin Jia, Junping Du, and Jun Zhang. On iterative learning algorithms for the formation control of nonlinear multi-agent systems. *Automatica*, 50 (1):291–295, 2014.

H Min, F Sun, S Wang, and H Li. Distributed adaptive consensus algorithm for networked euler-lagrange systems. *IET Control Theory & Application*, 5(1):145–154, 2011.

Kevin L Moore. *Iterative Learning Control for Deterministic Systems*. Springer-Verlag, London, 1993. Advances in Industrial Control.

Kevin L Moore, Yangquan Chen, and Hyo-Sung Ahn. Iterative learning control: A tutorial and big picture. In *Proceedings of the 45th IEEE Conference on Decision & Control*, pages 2352–2357, San Diego, CA, USA, 13-15 December 2006.

Luc Moreau. Stability of multiagent systems with time-dependent communication links. *IEEE Transactions on Automatic Control*, 50(2):169–182, 2005.

Peter B Moyle and Joseph J Cech. *Fishes: An Introduction to Ichthyology*. Benjamin Cummings, 5th edition, 2003.

Mikael Norrlof and Svante Gunnarsson. Time and frequency domain convergence properties in iterative learning control. *International Journal of Control*, 75(14):1114–1126, 2002.

Reza Olfati-Saber and Richard M. Murray. Consensus problems in networks of agents with switching topology and time-delays. *IEEE Transactions on Automatic Control*, 49(9):1520–1533, 2004.

Reza Olfati-Saber, J. Alex Fax, and Richard M. Murray. Consensus and cooperation in networked multi-agent systems. *Proceedings of the IEEE*, 95(1):215–233, 2007.

P R Ouyang, W J Zhang, and Madan M Gupta. An adaptive switching learning control method for trajectory tracking of robot manipulators. *Mechatronics*, 16:51–61, 2006.

Kwang-Hyun Park. An average operator-based pd-type iterative learning control for variable initial state error. *IEEE Transactions on Automatic Control*, 50(6):865–869, 2005.

Kwang-Hyun Park, Zeungnam Bien, and Dong-Hwan Hwang. A study on the robustness of a pid-type iterative learning controller against initial state error. *International Journal of Systems Science*, 30(1):49 – 59, 1999.

Wei Ren. Synchronization of coupled harmonic oscillators with local interaction. *Automatica*, 44(12):3195–3200, 2008a.

Wei Ren. On consensus algorithms for double integrator dynamics. *IEEE Transactions on Automatic Control*, 53(6):1503–1509, 2008b.

Wei Ren. Distributed leaderless consensus algorithms for networked eulerclagrange systems. *International Journal of Control*, 82(11):2137–2149, 2009.

Wei Ren and Randal W Beard. *Distributed Consensus in Multi-vehicle Cooperative Control*. Communication and Control Engineering Series. Springer-Verlag, London, 2008.

Wei Ren and Yongcan Cao. *Distributed Coordination of Multi-agent Networks*. Communication and Control Engineering Series. Springer-Verlag, London, 2011.

Wei Ren, Randal W Beard, and Ella M Atkins. Information consensus in multivehicle cooperative control. *IEEE Control Systems Magazine*, pages 71–82, April 2007.

Samer S Saab. On the p-type learning control. *Transactions on Automatic Control*, 39 (11):2298 – 2302, 1994.

Sanjib Kumar Sahoo, Sanjib Kumar Panda, and Jina-Xin Xu. Iterative learning-based high-performance current controller for switched reluctance motors. *IEEE Transactions on Energy Conversion*, 19(3):491 –498, 2004.

Jiantao Shi, Xiao He, Zidong Wang, and Donghua Zhou. Iterative consensus for a class of second-order multi-agent systems. *Journal of Intelligent & Robotic Systems*, 73 (1-4):655 – 664, 2014.

Jean-Jacques E Slotine and Weiping Li. *Applied Nonlinear Control*. Prentice Hall, 1991.

Eduardo D Sontag. *Input to state stability: Basic concepts and results*. Springer, 2006.

Marija Ilic Spong, Riccardo Marino, Sergei M Peresada, and David G Taylor. Feedback linearizing control of switched reluctance motors. *IEEE Transactions on Automatic Control*, 32(5):371 – 379, 1987.

Mark W Spong, Seth Hutchinson, and M Vidyasagar. *Robot Modeling and Control*. John Wiley & Sons Inc, 2006.

Mingxuan Sun and Danwei Wang. Iterative learning control with initial rectifying action. *Automatica*, 38(7):1177–1182, 2002.

Mingxuan Sun, Shuzhi Sam Ge, and Iven M Y Mareels. Adaptive repetitive learning control of robotic manipulators without the requirement for initial repositioning. *IEEE Transactions on Robotics*, 22(3):563–568, 2006.

V L Syrmos, C T Abdallah, P Dorato, and K Grigoriadis. Static output feedback - a survey. *Automatica*, 33(2):125–137, 1997.

Alireza Tahbaz-Salehi and Ali Jadbabaie. A necessary and sufficient condition for consensus over random networks. *IEEE Transactions on Automatic Control*, 53(3):791–795, 2008.

Abdelhamid Tayebi. Adaptive iterative learning control of robot manipulators. *Automatica*, 40:1195–1203, 2004.

Abdelhamid Tayebi and S Islam. Adaptive iterative learning control of robot manipulators: Experimental results. *Control Engineering Practice*, 14:843–851, 2006.

Long Wang and Feng Xiao. Finite-time consensus problems for networks of dynamic agents. *IEEE Transactions on Automatic Control*, 55(4):950–955, 2010.

Xiaoli Wang and Yiguang Hong. Finite-time consensus for multi-agent networks with second-order agent dynamics. In *Proceedings of the 17th IFAC World Congress*, pages 15185–15190, Seoul, Korea, 6-11 July 2008.

Youqing Wang, Furong Gao, and Francis J Doyle III. Survey on iterative learning control, repetitive control, and run-to-run control. *Journal of Process Control*, 19: 1589–1600, 2009.

Peter Wieland, Rodolphe Sepulchre, and Frank Allgower. An internal model principle is necessary and sufficient for linear output synchronization. *Automatica*, 47(3):1068–1074, 2011.

Chai Wah Wu. Synchronization and convergence of linear dynamics in random directed networks. *IEEE Transactions on Automatic Control*, 51(7):1207–1270, 2006.

Ji Xiang, Wei Wei, and Yanjun Li. Synchronized output regulation of linear networked systems. *IEEE Transactions on Automatic Control*, 54(6):1336–1341, 2009.

Lin Xiao and Stephen Boyd. Fast linear iterations for distributed averaging. *Systems & Control Letters*, 53(1):65–78, 2004.

Guangming Xie and Long Wang. Consensus control for a class of networks of dynamic agents: Fixed topology. In *Proceedings of the 44th IEEE Conference on Decision and Control and the European Control Conference 2005*, pages 96–101, Seville, Spain, 12-15 December 2005.

Jian-Xin Xu and Zhihua Qu. Robust iterative learning control for a class of nonlinear systems. *Automatica*, 34(8):983–988, 1998.

Jian-Xin Xu and Ying Tan. A composite energy function based learning control approach for nonlinear systems with time varying parametric uncertainties. *IEEE Transaction on Automatic Control*, 47(11):1940–1945, 2002a.

Jian-Xin Xu and Ying Tan. Robust optimal design and convergence properties analysis of iterative learning control approaches. *Automatica*, 38(11):1867–1880, 2002b.

Jian-Xin Xu and Ying Tan. *Linear and Nonlinear Iterative Learning Control*. Springer-Verlag, Germany, 2003. In series of Lecture Notes in Control and Information Sciences.

Jian-Xin Xu and Jing Xu. On iterative learning from different tracking tasks in the presence of time-varying uncertainties. *IEEE Transactions On Systems, Man, and Cybernetics – Part B: Cybernetics*, 34(1):589–597, 2004.

Jian-Xin Xu and Rui Yan. On initial conditions in iterative learning control. *IEEE Transaction on Automatic Control*, 50(9):1349–1354, 2005.

Jian-Xin Xu and Shiping Yang. Iterative learning based control and optimization for large scale systems. In *13th IFAC Symposium on Large Scale Complex Systems: Theory and Applications*, pages 74–81, Shanghai, China, 7-10 July 2013.

Jian-Xin Xu, Yangquan Chen, T H Lee, and S Yamamoto. Terminal iterative learning control with an application to rtpcvd thickness control. *Automatica*, 35(9):1535–1542, 1999.

Jian-Xin Xu, Shuang Zhang, and Shiping Yang. A hoim-based iterative learning control scheme for multi-agent formation. In *2011 IEEE International Symposium on Intelligent Control*, pages 218–423, Denver, CO, USA, 28-30 September 2011.

Shiping Yang and Jian-Xin Xu. Adaptive iterative learning control for multi-agent systems consensus tracking. In *IEEE International Conference on Systems, Man, and Cybernetics*, pages 2803–2808, COEX, Seoul, Korea, 14-17 October 2012.

Shiping Yang and Jian-Xin Xu. Multi-agent consensus tracking with input sharing by iterative learning control. In *The 13th European Control Conference*, pages 868–873, Strasbourg, France, 24-27 June 2014.

Shiping Yang, Jian-Xin Xu, and Deqing Huang. Iterative learning control for multi-agent systems consensus tracking. In *The 51st IEEE Conference on Decision and Control*, pages 4672–4677, Maui, Hawaii, USA, 10-13 December 2012.

Shiping Yang, Sicong Tan, and Jian-Xin Xu. Consensus based approach for economic dispatch problem in a smart grid. *IEEE Transactions on Power Systems*, 28(4):4416–4426, 2013.

Chenkun Yin, Jian-Xin Xu, and Zhongsheng Hou. A high-order internal model based iterative learning control scheme for nonlinear systems with time-iteration-varying parameters. *IEEE Transaction on Automatic Control*, 55(11):2665–2670, 2010.

Alpaslan Yufka, Osman Parlaktuna, and Metin Ozkan. Formation-based cooperative transportation by a group of non-holonomic mobile robots. In *Systems Man and Cybernetics (SMC), 2010 IEEE International Conference on*, pages 3300–3307, 2010.

Wenlin Zhang, Zheng Wang, and Yi Guo. Backstepping-based synchronization of uncertain networked lagrangian system. *International Journal of Systems Science*, 2012. doi: 10.1080/00207721.2012.669869.

Xiao-Dong Zhang. The laplacian eigenvalues of graphs: A survey. In *arXiv:math.OC/arXiv:1111.2897v1*, 2011.

Ya Zhang and Yu-Ping Tian. Consentability and protocol design of multi-agent systems with stochastic switching topology. *Automatica*, 45:1195–1201, 2009.

Ke Ming Zhou and John C Doyle. *Essentials of Robust Control*. Prentice Hall, Upper Saddle River, New Jersy, 1998.

Appendix A

Graph Theory Revisit

Let $\mathscr{G} = (\mathscr{V}, \mathscr{E})$ be a weighted directed graph with the vertex set $\mathscr{V} = \{1, 2, \ldots, N\}$ and edge set $\mathscr{E} \subseteq \mathscr{V} \times \mathscr{V}$. Let $\mathscr{V}$ also be the index set representing the follower agents in the systems. A direct edge from k to j is denoted by an ordered pair $(k, j) \in \mathscr{E}$, which means that agent j can receive information from agent k. The neighborhood of the kth agent is denoted by the set $\mathscr{N}_k = \{j \in \mathscr{V} | (j,k) \in \mathscr{E}\}$. $\mathscr{A} = (a_{k,j}) \in \mathbb{R}^{N \times N}$ is the weighted adjacency matrix of $\mathscr{G}$. In particular, $a_{k,k} = 0$, $a_{k,j} = 1$ if $(j,k) \in \mathscr{E}$, and $a_{k,j} = 0$ otherwise[1]. The in-degree of vertex k is defined as $d_k^{in} = \sum_{j=1}^{N} a_{k,j}$, and the Laplacian of $\mathscr{G}$ is defined as $L = \mathscr{D} - \mathscr{A}$, where $\mathscr{D} = \mathrm{diag}(d_1^{in}, \ldots, d_N^{in})$. The Laplacian of an undirected graph is symmetric, whereas the Laplacian of a directed graph is asymmetric in general. An undirected graph is said to be connected if there is a path[2] between any two vertices. A spanning tree is a directed graph, whose vertices have exactly one parent except for one vertex, which is called the root who has no parent. We say that a graph contains or has a spanning tree if $\mathscr{V}$ and a subset of $\mathscr{E}$ can form a spanning tree.

[1]Undirected graph is a special case of directed graph, satisfying $a_{k,j} = a_{j,k}$.

[2]A path between vertices p and q is a sequence $(p = j_1, \ldots, j_l = q)$ of distinct vertices such that $(j_k, j_{k+1}) \in \mathscr{E}, \forall 1 \leq k \leq l-1$.

Important Properties of Laplacian Matrix:

- 0 is an eigenvalue of L and $\mathbf{1}$ is the associated eigenvector, namely, the row sum of Laplacian matrix is zero.
- If $\mathscr{G}$ has a spanning tree, the eigenvalue 0 is algebraically simple and all other eigenvalues have positive real parts.
- If $\mathscr{G}$ is strongly connected, then there exists a positive column vector $\mathbf{w} \in \mathbb{R}^N$ such that $\mathbf{w}^T L = 0$.

Furthermore, if $\mathscr{G}$ is undirected and connected, then L is symmetric and has following additional properties.

- $\mathbf{x}^T L\mathbf{x} = \frac{1}{2}\sum_{i,j=1}^{N} a_{ij}(x_i - x_j)^2$ for any $\mathbf{x} = [x_1, x_2, \ldots, x_N]^T \in \mathbb{R}^N$, and therefore L is positive semi-definite and all eigenvalues are positive except one zero eigenvalue.
- The second smallest eigenvalue of L, which is denoted by $\lambda_2(L) > 0$, is called the algebraic connectivity of $\mathscr{G}$. It determines the convergence rate of classic consensus algorithm.
- The algebraic connectivity

$$\lambda_2(L) = \inf_{\mathbf{x}\neq 0, \mathbf{1}^T\mathbf{x}=0} \frac{\mathbf{x}^T L\mathbf{x}}{\mathbf{x}^T\mathbf{x}},$$

  and therefore, if $\mathbf{1}^T\mathbf{x} = 0$, then $\mathbf{x}^T L\mathbf{x} \geq \lambda_2(L)\mathbf{x}^T\mathbf{x}$.

Appendix B

Detailed Proofs

B.1 Proof of Proposition 2.1

By Schur triangularization theorem (Horn and Johnson, 1985, pp. 79), there is a unitary matrix U and an upper triangular matrix Δ with diagonal entries being the eigenvalues of M, such that

$$\Delta = U^*MU,$$

where * denotes the conjugate transpose.

Let $Q = \mathrm{diag}(\alpha, \alpha^2, \ldots, \alpha^n),\ \alpha \neq 0$, and set $S = QU^*$. So S is nonsingular. Now define a matrix norm $|\cdot|_S$ (Horn and Johnson, 1985, pp.296) such that

$$|M|_S = |SMS^{-1}|,$$

where $|\cdot|$ can be any l_p vector norm induced matrix norm.

Compute SMS^{-1} explicitly, we can obtain

$$SMS^{-1} = \begin{bmatrix} \lambda_1 & \alpha^{-1}\delta_{1,2} & \alpha^{-2}\delta_{1,3} & \cdots & \alpha^{-n+1}\delta_{1,n} \\ 0 & \lambda_2 & \alpha^{-1}\delta_{2,3} & \cdots & \alpha^{-n+2}\delta_{2,n} \\ 0 & 0 & \lambda_3 & \cdots & \alpha^{-n+3}\delta_{3,n} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & 0 & \lambda_n \end{bmatrix},$$

where λ_i is an eigenvalue of M and $\delta_{i,j}$ is the (i,j)th entry of Δ.

Therefore, $|M|_S$ can be computed as below,

$$\begin{aligned} |M|_S &= \max_{|\mathbf{z}|=1} |SMS^{-1}\mathbf{z}| \\ &= \max_{|\mathbf{z}|=1} |M_0\mathbf{z} + E(\alpha)\mathbf{z}| \\ &\leq \max_{|\mathbf{z}|=1} |M_0\mathbf{z}| + \max_{|\mathbf{z}|=1} |E(\alpha)\mathbf{z}|, \end{aligned} \tag{B.1}$$

where $M_0 = \mathrm{diag}(\lambda_1, \lambda_2, \ldots, \lambda_n)$ and

$$E(\alpha) = \begin{bmatrix} 0 & \alpha^{-1}\delta_{1,2} & \alpha^{-2}\delta_{1,3} & \cdots & \alpha^{-n+1}\delta_{1,n} \\ 0 & 0 & \alpha^{-1}\delta_{2,3} & \cdots & \alpha^{-n+2}\delta_{2,n} \\ 0 & 0 & 0 & \cdots & \alpha^{-n+3}\delta_{3,n} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix}.$$

It is easy to verify that

$$\begin{aligned} \max_{|\mathbf{z}|=1} |M_0\mathbf{z}| &= \max_{|\mathbf{z}|=1} \left| \begin{bmatrix} \lambda_1 z_1 & \lambda_2 z_2 & \cdots & \lambda_n z_n \end{bmatrix}^T \right| \\ &\leq \max_{j=1,2,\ldots,n} |\lambda_j| \max_{|\mathbf{z}|=1} |\mathbf{z}| = \rho(M). \end{aligned} \tag{B.2}$$

Define the last term in (B.1) as a function of α, $g(\alpha) = \max_{|\mathbf{z}|=1} |E(\alpha)\mathbf{z}|$. As $g(\alpha)$ is a continuous function of α and $\lim_{\alpha \to \infty} g(\alpha) = 0$, therefore, for any $\varepsilon = (1-\rho(M))/2$, there exists an α^* such that $g(\alpha) < \varepsilon$ for all $\alpha > \alpha^*$. Substituting (B.2) to (B.1) and choosing $\alpha > \alpha^*$, we have

$$|M|_S \leq \rho(M) + \varepsilon < (1+\rho(M))/2 < 1.$$

Therefore, we can conclude that $\lim_{k\to\infty} (|M|_S)^k = 0$. ■

B.2 Proof of Lemma 2.1

The feasible region of the optimization problem

$$\min_{\gamma\in\mathbb{R}} \max_{\alpha_1 \leq a \leq \sqrt{a^2+b^2} \leq \alpha_2} |1-\gamma(a+jb)|$$

can be classified into three regions according to γ.

1. $\gamma > 0$, denoting $J_1 \triangleq \min_{\gamma>0} \max_{\alpha_1 \leq a \leq \sqrt{a^2+b^2} \leq \alpha_2} |1-\gamma(a+jb)|$, the boundary of $a+jb$ can be seen in Figure. B.1.

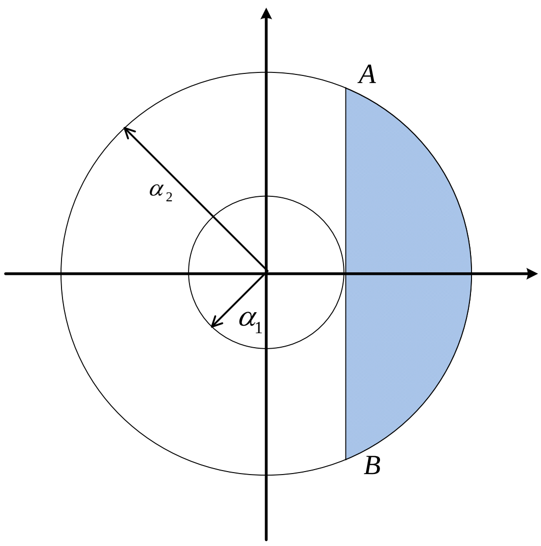

Figure B.1: The boundary of complex parameter $a+jb$.

According to Proposition 2.4, $|1-\gamma(a+jb)|$ reaches its maximum value at the boundary of the compact set (the shadow area in Figure. B.1). The maximum value in chord $\overline{AB}$ is either at point $A = \left(\alpha_1, \sqrt{\alpha_2^2 - \alpha_1^2}\right)$ or at point $B = \left(\alpha_1, -\sqrt{\alpha_2^2 - \alpha_1^2}\right)$, because the imaginary parts of these two points reach the maximum value in $\overline{AB}$ while the real parts in $\overline{AB}$ retain the constant α_1.

$$\begin{aligned}
& \max_{(a,b)\in\overline{AB}} |1-\gamma(a+jb)| \\
= & \sqrt{(1-\gamma\alpha_1)^2 + \gamma^2\left(\sqrt{\alpha_2^2-\alpha_1^2}\right)^2} \\
= & \sqrt{1-2\gamma\alpha_1+\gamma^2\alpha_2^2}. \qquad \text{(B.3)}
\end{aligned}$$

Along the arc $\widehat{AB}$, $\forall\gamma > 0$, the real part $a = \alpha_2 \sin(\theta)$ and the imaginary part $b = \alpha_2 \cos(\theta)$, with $-\bar{\theta} \le \theta \le \bar{\theta}$ and $\bar{\theta} \triangleq \arccos\left(\frac{\alpha_1}{\alpha_2}\right)$. The maximum value can be calculated as

$$\begin{aligned}
& \max_{(a,b)\in\widehat{AB}} |1-\gamma(a+jb)| \\
= & \max_{-\bar{\theta}\le\theta\le\bar{\theta}} \sqrt{(1-\gamma\alpha_2\cos(\theta))^2 + \gamma^2\alpha_2^2\sin^2(\theta)} \\
= & \max_{-\bar{\theta}\le\theta\le\bar{\theta}} \sqrt{1-2\gamma\alpha_2\cos(\theta)+\gamma^2\alpha_2^2} \\
= & \sqrt{1-2\gamma\alpha_2\frac{\alpha_1}{\alpha_2}+\gamma^2\alpha_2^2} \\
= & \sqrt{1-2\gamma\alpha_1+\gamma^2\alpha_2^2}. \qquad \text{(B.4)}
\end{aligned}$$

Consequently when $\gamma = \frac{\alpha_1}{\alpha_2^2}$ we have

$$\begin{aligned}
J_1 &= \min_{\gamma>0} \sqrt{1-2\gamma\alpha_1+\gamma^2\alpha_2^2} \\
&= \min_{\gamma>0} \sqrt{\alpha_2^2\left(\gamma - \frac{\alpha_1}{\alpha_2^2}\right)^2 + 1 - \frac{\alpha_1^2}{\alpha_2^2}} \\
&= \frac{\sqrt{\alpha_2^2-\alpha_1^2}}{\alpha_2}. \qquad \text{(B.5)}
\end{aligned}$$

2. $\gamma = 0, J_2 \triangleq \min_{\gamma=0} \max_{\alpha_1\le a\le\sqrt{a^2+b^2}\le\alpha_2} |1-\gamma(a+jb)| = 1.$

3. $\gamma < 0, J_3 \triangleq \min_{\gamma<0} \max_{\alpha_1\le a\le\sqrt{a^2+b^2}\le\alpha_2} |1-\gamma(a+jb)| > 1.$

Therefore, $\min_{\gamma\in\mathbb{R}} \max_{\alpha_1\leq a\leq\sqrt{a^2+b^2}\leq\alpha_2} |1-\gamma(a+jb)| = \min\{J_1,J_2,J_3\} = \dfrac{\sqrt{\alpha_2^2-\alpha_1^2}}{\alpha_2}$. ■

B.3 Proof of Theorem 6.1

From Assumption 6.2 and Lemma 6.1, we can get that H is a symmetric positive definite matrix, and

$$(\min_j \underline{b}_j)I \leq B \leq (\max_j \overline{b}_j)I.$$

Therefore, $E_i(t)$ is a nonnegative function.

The proof consists of two parts. In Part A, the difference of E_i between two consecutive iterations is calculated, and convergence of tracking error is shown in Part B.

Part A: Difference of E_i

The difference of $E_i(t)$ is defined as

$$\begin{aligned}
\Delta E_i(t) &= E_i(t) - E_{i-1}(t), \\
&= V_i(\mathbf{e}_i) - V_{i-1}(\mathbf{e}_{i-1}) + \frac{1}{2\kappa}\int_0^t \text{Trace}\left((\tilde{\Theta}_i(\tau))^T B\tilde{\Theta}_i(\tau)\right) d\tau \\
&\quad -\frac{1}{2\kappa}\int_0^t \text{Trace}\left((\tilde{\Theta}_{i-1}(\tau))^T B\tilde{\Theta}_{i-1}(\tau)\right) d\tau \qquad \text{(B.6)}
\end{aligned}$$

Assumption 6.3 indicates that $V_i(\mathbf{e}_i(0)) = 0$. The first term in (B.6) becomes

$$\begin{aligned}
V_i(\mathbf{e}_i) &= \int_0^t \dot{V}_i(\mathbf{e}_i)\, d\tau + V_i(\mathbf{e}_i(0)) \\
&= \int_0^t \dot{V}_i(\mathbf{e}_i)\, d\tau. \qquad \text{(B.7)}
\end{aligned}$$

Noting that $\dot{V}_i(\mathbf{e}_i) = \mathbf{e}_i^T H\dot{\mathbf{e}}_i = \varepsilon_i^T \dot{\mathbf{e}}_i$, together with the closed loop error dynamics (6.13), it yields

$$\dot{V}_i = \varepsilon_i^T B\tilde{\Theta}_i\xi(t,\mathbf{x}_i) + \varepsilon_i^T\left(\eta_d - \eta(t,\mathbf{x}_i)\right) - \varepsilon_i^T\left(\gamma I + \left(\overline{\Phi}(\mathbf{x}_i)\right)^2\right)\varepsilon_i. \qquad \text{(B.8)}$$

Since $\eta_j(t, x_{i,j})$ satisfies Assumption 6.1, noticing (6.4), we have

$$\begin{aligned}
& \left|\varepsilon_i^T \left(\eta_d - \eta(t, \mathbf{x}_i)\right)\right| \\
\leq \;& \sum_{j=1}^{N} \phi_j(x_d, x_{i,j}) |\varepsilon_{i,j}| \cdot |e_{i,j}| \leq \varepsilon_i^T \left(\overline{\Phi}(\mathbf{x}_i)\right)^2 \varepsilon_i + \frac{1}{4} \mathbf{e}_i^T \mathbf{e}_i \\
= \;& \varepsilon_i^T \left(\overline{\Phi}(\mathbf{x}_i)\right)^2 \varepsilon_i + \frac{1}{4} \varepsilon_i^T H^{-2} \varepsilon_i \\
\leq \;& \varepsilon_i^T \left(\overline{\Phi}(\mathbf{x}_i)\right)^2 \varepsilon_i + \frac{1}{4\underline{\sigma}(H)^2} \varepsilon_i^T \varepsilon_i. \qquad \text{(B.9)}
\end{aligned}$$

Substituting (B.9) to (B.8), using the convergence condition (6.15) leads to

$$\begin{aligned}
\dot{V}_i \leq \;& \varepsilon_i^T B \tilde{\Theta}_i \xi(t, \mathbf{x}_i) + \varepsilon_i^T \left(\overline{\Phi}(\mathbf{x}_i)\right)^2 \varepsilon_i + \frac{1}{4\underline{\sigma}(H)^2} \varepsilon_i^T \varepsilon_i - \varepsilon_i^T \left(\gamma I + \left(\overline{\Phi}(\mathbf{x}_i)\right)^2\right) \varepsilon_i \\
\leq \;& -\alpha \varepsilon_i^T \varepsilon_i + \varepsilon_i^T B \tilde{\Theta}_i \xi(t, \mathbf{x}_i) \qquad \text{(B.10)}
\end{aligned}$$

Combining the third term and the fourth term in (B.6) yields

$$\begin{aligned}
& \text{Trace}\left(\tilde{\Theta}_i^T B \tilde{\Theta}_i\right) - \text{Trace}\left(\tilde{\Theta}_{i-1}^T B \tilde{\Theta}_{i-1}\right) \\
= \;& \text{Trace}\left((\hat{\Theta}_{i-1} - \hat{\Theta}_i)^T B (2\Theta_i - \hat{\Theta}_i - \hat{\Theta}_{i-1})\right) \\
= \;& \text{Trace}\left((\hat{\Theta}_{i-1} - \hat{\Theta}_i)^T B (2\Theta_i - 2\hat{\Theta}_i + \hat{\Theta}_i - \hat{\Theta}_{i-1})\right) \\
= \;& \text{Trace}\left((\hat{\Theta}_{i-1} - \hat{\Theta}_i)^T B (2\tilde{\Theta}_i + \hat{\Theta}_i - \hat{\Theta}_{i-1})\right) \\
= \;& -\text{Trace}\left((\hat{\Theta}_{i-1} - \hat{\Theta}_i)^T B (\hat{\Theta}_{i-1} - \hat{\Theta}_i)\right) + 2\text{Trace}\left((\hat{\Theta}_{i-1} - \hat{\Theta}_i)^T B \tilde{\Theta}_i\right). \qquad \text{(B.11)}
\end{aligned}$$

From equations (B.6), (B.7), (B.10), and (B.11), one has

$$\begin{aligned}
\Delta E_i \leq \;& -\frac{1}{2} \mathbf{e}_{i-1}^T H \mathbf{e}_{i-1} + \int_0^t -\alpha \varepsilon_i^T \varepsilon_i d\tau + \int_0^t \varepsilon_i^T B \tilde{\Theta}_i \xi(t, \mathbf{x}_i) d\tau \\
& - \int_0^t \frac{1}{2\kappa} \text{Trace}\left((\hat{\Theta}_{i-1} - \hat{\Theta}_i)^T B (\hat{\Theta}_{i-1} - \hat{\Theta}_i)\right) d\tau \\
& + \int_0^t \frac{1}{\kappa} \text{Trace}\left((\hat{\Theta}_{i-1} - \hat{\Theta}_i)^T B \tilde{\Theta}_i\right) d\tau. \qquad \text{(B.12)}
\end{aligned}$$

From parameter updating rule (6.11), it can be shown that

$$\varepsilon_i^T B \tilde{\Theta}_i \xi(t, \mathbf{x}_i) + \frac{1}{\kappa} \text{Trace}\left((\hat{\Theta}_{i-1} - \hat{\Theta}_i)^T B \tilde{\Theta}_i\right) = 0,$$

and Trace $\left((\hat{\Theta}_{i-1}-\hat{\Theta}_i)^T B(\hat{\Theta}_{i-1}-\hat{\Theta}_i)\right) \geq 0$.

Therefore, (B.12) becomes

$$\Delta E_i \leq -\frac{1}{2}\mathbf{e}_{i-1}^T H \mathbf{e}_{i-1} \leq 0. \tag{B.13}$$

Part B: Convergence of $e_{i,j}$

If the boundedness of E_1 is proven, following the same steps in Xu and Tan (2002a), we can show the point-wise convergence of $e_{i,j}$. Taking derivative of E_1, together with (B.10), simple manipulations lead to

$$\begin{aligned}\dot{E}_1 &\leq -\alpha\varepsilon_1^T\varepsilon_1 + \varepsilon_1^T B(\Theta-\hat{\Theta}_1)\xi(t,\mathbf{x}_1) + \frac{1}{2\kappa}\text{Trace}\left((\Theta-\hat{\Theta}_1)^T B(\Theta-\hat{\Theta}_1)\right)\\ &\leq \frac{1}{2\kappa}\text{Trace}\left(\Theta^T B\Theta\right).\end{aligned}$$

Θ is a finite and continuous signal, hence, $\dot{E}_1$ is bounded in the interval $[0,T]$. Subsequently, E_1 is bounded in the finite-time interval $[0,T]$. ■

B.4 Proof of Corollary 6.1

The proof is completed by evaluating the CEF defined in (6.14) at the time $t = T$.

By using Assumption 6.4, $V_i(0) = V_{i-1}(T)$, the difference between $V_i(T)$ and $V_{i-1}(T)$ can be written as

$$\begin{aligned}\Delta V_i(T) &= \int_0^T \dot{V}_i(\tau)d\tau + V_i(0) - V_{i-1}(T)\\ &= \int_0^T \dot{V}_i(\tau)d\tau.\end{aligned}$$

By following the similar proof of Theorem 6.1, eventually, we can obtain that

$$\begin{aligned}E_i(T) &= E_1(T) + \sum_{k=2}^{i}\Delta E_k(T)\\ &\leq E_1(T) - \sum_{k=2}^{i}\alpha\int_0^T (\mathbf{e}_k(\tau))^T H^2\mathbf{e}_k(\tau)d\tau.\end{aligned}$$

Since $E_1(T)$ is bounded, $E_i(T)$ is nonnegative, and H^2 is positive definite, it follows that

$$\lim_{i\to\infty}\int_0^T (\mathbf{e}_i(\tau))^T \mathbf{e}_i(\tau)\, d\tau = 0.$$

This completes the proof. ∎

Appendix C

Author's Publications

The author has contributed to the following publications:

Journal Papers:

[1] S. Yang and J.-X. Xu, "Improvements on 'a new framework of consensus protocol design for complex multi-agent systems'," *Systems & Control Letters*, vol. 61, no. 9, pp. 945–949, 2012.

[2] S. Yang, S. Tan, and J.-X. Xu, "Consensus based approach for economic dispatch problem in a smart grid," *IEEE Transactions on Power Systems*, vol. 28, no. 4, pp. 4416–4426, 2013.

[3] S. Yang and J.-X. Xu, "Leader-follower synchronization for networked lagrangian systems with uncertainties: A learning approach," *International Journal of Systems Science*, 2014. in press.

[4] S. Yang, J.-X. Xu, D. Huang, and Y. Tan, "Optimal iterative learning control design for multi-agent systems consensus tracking," *Systems & Control Letters*, vol. 69, pp. 80–89, 2014.

[5] S. Yang, J.-X. Xu, D. Huang, and Y. Tan, "Synchronization of heterogeneous agent systems by adaptive iterative learning control," *Asian Journal of Control*, 2014. Submitted.

[6] S. Yang and J.-X. Xu, "Multi-agent consensus tracking by iterative learning control under iteration-varying graph," *International Journal of Robust and Nonlinear Control*, 2014. Submitted.

[7] D. Huang, J.-X. Xu, S. Yang, and X. Jin, "Observer based repetitive learning control for a class of nonlinear systems with non-parametric uncertainties," *International Journal of Robust and Nonlinear Control*, 2014. in press.

[8] Q. Zhu, J.-X. Xu, S. Yang, and G.-D. Hu, "Adaptive backstepping repetitive learning control design for nonlinear discrete-time systems with periodic uncertainties," *International Journal of Adaptive Control and Signal Processing*, 2014. in press.

Conference Papers:

[1] J.-X. Xu, S. Zhang, and S. Yang, "A hoim-based iterative learning control scheme for multi-agent formation," in *2011 IEEE International Symposium on Intelligent Control*, (Denver, CO, USA), pp. 218–423, 28-30 September 2011.

[2] J.-X. Xu and S. Yang, "Decentralized coordination control mas with workload learning," in *2011 IEEE International Symposium on Intelligent Control*, (Denver, CO, USA), pp. 212–417, 28-30 September 2011.

[3] S. Yang, J.-X. Xu, and D. Huang, "Iterative learning control for multi-agent systems consensus tracking," in *The 51st IEEE Conference on Decision and Control*, (Maui, Hawaii, USA), pp. 4672–4677, 10-13 December 2012.

[4] S. Yang and J.-X. Xu, "Adaptive iterative learning control for multi-agent systems consensus tracking," in *IEEE International Conference on Systems, Man, and Cybernetics*, (COEX, Seoul, Korea), pp. 2803–2808, 14-17 October 2012.

[5] S. Yang, Y. Tan, and J.-X. Xu, "On iterative learning control for synchronization of mimo heterogeneous systems," in *Australian Control Conference*, (Perth, Australia), pp. 379–384, 4-5 November 2013.

[6] J.-X. Xu and S. Yang, "Iterative learning based control and optimization for large scale systems," in *13th IFAC Symposium on Large Scale Complex Systems: Theory and Applications*, (Shanghai, China), pp. 74–81, 7-10 July 2013.

[7] S. Yang, J.-X. Xu, and M. Yu, "An iterative learning control approach for synchronization of multi-agent systems under iteration-varying graph," in *The 52nd IEEE Conference on Decision and Control*, (Florence, Italy), pp. 6682–6687, 10-13 December 2013.

[8] S. Yang and J.-X. Xu, "Multi-agent consensus tracking with input sharing by iterative learning control," in *The 13th European Control Conference*, (Strasbourg, France), pp. 868–873, 24-27 June 2014.

[9] S. Yang, J.-X. Xu, and Q. Ren, "Multi-agent consensus tracking with initial state error by iterative learning control," in *The 11th IEEE International Conference on Control & Automation*, (Taichung, Taiwan), pp. 625–630, 18-20 June 2014.