Distributed Algorithms for Interacting Autonomous Agents

Weiguo Xia

The research described in this dissertation has been carried out at the Faculty of Mathematics and Natural Sciences, University of Groningen, the Netherlands.

The research reported in this dissertation is part of the research program of the Dutch Institute of Systems and Control (DISC). The author has successfully completed the educational program of DISC.

Printed by Ipskamp Drukkers B.V.
Enschede, the Netherlands

RIJKSUNIVERSITEIT GRONINGEN

Distributed Algorithms for Interacting Autonomous Agents

Proefschrift

ter verkrijging van het doctoraat in de
Wiskunde en Natuurwetenschappen
aan de Rijksuniversiteit Groningen
op gezag van de
Rector Magnificus, dr. E. Sterken,
in het openbaar te verdedigen op
vrijdag 28 juni 2013
om 14.30 uur

door

Weiguo Xia

geboren op 26 december 1984
te Jiangsu, China

Promotor:	Prof. dr. ir. J.M.A. Scherpen
Copromotor:	Dr. M. Cao
Beoordelingscommissie:	Prof. dr. A.S. Morse Prof. dr. V.D. Blondel Prof. dr. C. De Persis

ISBN: 978-90-367-6231-1

Contents

Acknowledgments

This thesis summarizes my research in the past four years within the group of Discrete Technology and Production Automation (DTPA) at the University of Groningen. I am pleased to be a member of the multi-cultural group of DTPA and am grateful to the group members and the support staff for their help.

First of all, I would like to thank my supervisor Ming Cao for his supervision. I have learnt a lot under his guidance on various aspects of scientific research. His enthusiasm and patience have encouraged me very much especially at times when I got stuck in my research. I appreciate his understanding and support during the difficult times throughout the past four years.

I want to thank my promotor Jacquelien Scherpen, who read the manuscript and raised a number of suggestions for improvement.

I would like to thank the reading committee members, Prof. Claudio De Pesis, Prof. Vincent D. Blondel, and Prof. A. Stephen Morse, for reading the manuscript and providing valuable comments.

In addition, I am grateful to Wenwu Yu, who informed me this Ph.D. student position before I came to the University of Groningen. It is my pleasure to thank my colleagues and friends for the valuable conversations and friendship: Gunn Larsen, Bayu Jayawardhana, Daniel Dirksz, Manuel Mazo Jr., Marko Seslija, Dustano del Puerto Flores, Ewoud Vos, Tao Liu, Shodhan Rao, James Riehl, Desti Alkano, Héctor Garcia de Marina Peinado, Robert Huisman, Hui Liu, Mauricio Munos Arias, Ruiyue Ouyang, Pouria Ramazi, Chen Wang, Fan Zhang, Shuo Zhang, Jianlei Zhang, Matin Jafarian, Sebastian Trip, Kanat Camlibel, Jieqiang Wei, Nima Monshizadeh.

Last but not least, I would like to thank my family for their endless love and support.

List of symbols

Chapter 1

Introduction

This thesis is concerned with distributed algorithms for interacting autonomous agents. In this thesis, we study several distributed algorithms that drive a group of agents to reach an agreement on the value of a variable of common interest or to split into two or more clusters. This chapter introduces some background information on emergent collective behavior that arises in natural and man-made systems and distributed algorithms developed for complex multi-agent systems. The motivation for the research in this thesis and the contributions are provided and the outline of the thesis follows.

1.1 Background

In this section, we give an introduction on the research of collective behavior in complex multi-agent systems and the design of distributed algorithms operating on these systems.

1.1.1 Collective behavior in complex multi-agent systems

As a class of collective behavior of groups of interacting units, synchronization has been discovered widely in natural, social and engineered networks and systems [93]. For example, fireflies flash in unison, audiences clap synchronously after an excellent performance, and coupled metronomes oscillate in phase. The discovery of this synchronization phenomenon can be traced back to as early as 1665, when a Dutch physicist Christiaan Huygens observed the synchrony of two clocks hanged on two planks which lied on top of two chairs [93]. The two pendulums gradually oscillated out of phase after some time and they kept swinging in this fashion from then on. After conducting a number of experiments, Huygens found out the cause of the synchrony of the clocks. The clocks were interacting through tiny vibrations of the planks and the chairs, which were caused by the swing of the pendulums. Since then, huge efforts have been devoted to the study of the synchronization phenomena from diverse disciplines like mathematics, physics, sociology, engineering and so on [7].

Besides the synchronization phenomena, various intriguing animal group behaviors have received considerable attention, such as schools of fish, flocks of birds, herds of cattle and so on [93]. People are more and more getting fascinated by and also try to unveil the mechanisms behind these collective behaviors in complex networks. A common character of these emerging collective behaviors is that the individual unit in the network does not have global information of all the units, instead, each one only receives limited information from its neighbors or peers nearby. Each individual unit in the network takes actions based on the local information she receives, while in the group level, surprisingly some collective behavior appears. It is challenging to investigate how the individual agent dynamics and the network structure interact with each other and jointly lead to global emergent collective behavior, which has been the central topic for network science.

In computer sciences, Reynolds developed programs to simulate the motion of a flock of birds [83]. Each individual bird has local perception of its surrounding environment and so has local information of where his peers are going, and reacts independently. Three rules operating on each individual bird lead to the emergence of flocking: (1) separation, avoid collisions with neighbors; (2) alignment, steer towards the average heading of nearby flockmates; (3) cohesion, steer towards the average position of nearby flockmates [83]. Mathematical models capturing these three rules were later proposed to theoretically explain the emergence of flocking of birds [71]. These shed light on better understandings of animal group behaviors and also may find potential applications in the formation flight of autonomous vehicles.

In [97], Vicsek *et al*. conducted experiments on self-propelled particles that move with constant absolute speed but adapting their moving directions to the average of those particles in their neighborhoods under some perturbations. This simple nearest neighbor rule successfully drives all the agents to move in the same direction, although there is no central station that broadcasts coordination commands and the neighborhood of each agent may change with time. This has inspired researchers from mathematics and engineering to explore the intrinsic mechanisms that induce the cohesive behavior [47] and also inspired the research on distributed control of multi-agent systems.

The last decades have witnessed major advances in the understanding of these collective behaviors, especially the synchronization phenomena of coupled dynamical systems [103, 102]. With the aid of supercomputers, people are enabled to handle vast amount of data, which makes it possible to analyze large-scale complex networks [24]. For a very long time, the theory of Erdös-Rényi random graphs has dominated the research of graph theory, which is a key tool for networks study. However, most of the real-world networks cannot be modeled as random networks. For example, whether there is a flight between two cities, whether two persons are friends, are not completely random. Two seminal papers published a decade ago led to significant advances in the field of networks studies. One proposed the well-known small-world network models that can be highly clustered but still have small path lengths [99] and the other revealed the fundamental feature of the scale-free property of various complex networks [9]. A number of research efforts have been following this research line and further investigating the collective behaviors that emerge in these networks.

In sociology, different mathematical models have been constructed to study the evolution of the opinions of a group of interacting individuals. Some of the models concerned are linear and people are more focused on the consensus problem and try to find out how to reach it [29]. Recently, more and more nonlinear models have been constructed to characterize the opinion dynamics in social communities. One opinion dynamics model called Krause model that has attracted great attention, considers that each agent in the network has a "confidence bound" [43]. When updating the opinions, each agent only takes into account those agents' opinions that differ no more than the confidence bound from his own opinion. The neighborhood of each agent thus changes with time and also depends on the state of the system. This model finally shows that the agents' states reach a consensus or polarize or become fragmented. These observed simulations results in the Krause model have been analyzed in different settings in [11, 12, 68].

In the study of social networks, structural balance theory that was proposed in 1950s plays an important role [22]. A structurally balanced network is a network that can be divided into two opposing factions, in which all links inside each faction are positive and all links between individuals in different factions are negative. Typical examples include two-party political systems, Western Bloc and Eastern Bloc during the Cold War and so on. Computations also have shown that lots of large-scale online social networks are structurally balanced [35]. The theory of structural balance, from which one can tell a given social network is balanced or not, is a static theory. However, the relationship between people and the structure of the network are changing with time. People are also trying to find out how a network may dynamically evolve into a structurally balanced state [64].

1.1.2 Distributed algorithms for multi-agent systems

Along with the growing interest in understanding the intriguing collective animal group behaviors, in the control community, there is also an emerging interest in studying distributed algorithms for multi-agent systems. Control of a single system has been well studied in the control society and various control theories have been developed, such as proportional-integral-derivative control, adaptive control, robust control and so on. Recently, there is an emerging trend to study distributed control of multiple interconnected systems. There are obvious advantages using several interconnected systems over a single complicated system in many practical applications, such as scalability, robustness, and flexibility.

Another reason that the study on distributed control of multi-agent systems has received considerable attention in recent years is due to their broad applications in sensor networks, unmanned aerial vehicles (UAVs), robotic teams, and so on [82]. The main objective in these applications is to control these agents in a group to accomplish some global task cooperatively by sharing only local information with their neighbors. This distributed control methodology differs tremendously from the traditional centralized control approach, in which a central station is required to control the agents in a network. Instead, no central station is available in a distributed control system. However, this distributed fashion also brings great challenge to the analysis of distributed control systems due to the fact that a number of systems are involved and also due to practical constraints, such as limited sensing capabilities, unreliable communication channels, asynchronous effects and so on. The global dynamical behavior of the interconnected system is thus highly nonlinear and difficult to predict. It is tempting and challenging to carry out research on the interplay between the local interactions among the agents and the emergent collective behavior of the integrated system.

The study on distributed control of multi-agent networks in control community can be traced back at least to 1980s. Efficient load balancing algorithms in distributed computation have been constructed successfully using ideas of distributed averaging that can be modeled using stochastic matrices [10]. Recently, there has been a resurgence of research interest in the study of distributed algorithms of multi-agent systems [47, 73, 69, 81, 72]. In [47], simple nearest neighbor rules have been used to successfully cause a group of agents to reach an agreement on their moving directions. Some connectivity conditions, which require that the agents are connected to each other throughout the network across each time interval, are constructed to guarantee the convergence. Rigorous mathematical proof of the convergence of the headings of all the agents to the same direction has been established by making use of matrix theory, graph theory and dynamical system theory. This also explains the simulation results conducted in [97], where the directions of the self-driven particles become the same in the presence of perturbations. In [81, 69], the connectivity assumption has been relaxed to the condition that the union graph across each time interval contains a directed spanning tree. This condition means that there always exists an agent that can influence all the other agents directly or indirectly in the network across each time interval, which guarantees the convergence of the overall system. The notions related to connectivity of a graph will be introduced in Chapter 2.

Following this research line, significant efforts have been devoted to the study of distributed algorithms along different research directions. Several active research directions include distributed consensus/clustering algorithms, distributed optimization, distributed formation control, distributed estimation and control, and so on [82]. Distributed consensus problem has been a fundamental and benchmark problem in multi-agent coordination. This thesis starts from studying an asynchronous implementation of a distributed averaging algorithm, and then continues investigating clustering algorithms in multi-agent systems, which includes the previous averaging algorithm as a special case. It is followed by determination of clock synchronization errors in distributed networks.

1.2 Motivation and contributions

Matrix theory, graph theory and dynamical system theory are powerful tools in the analysis of distributed coordination algorithms for multi-agent systems. Convergence of products of stochastic matrices has proven to be critical in establishing the effectiveness of distributed coordination algorithms. The study of the convergence of products of non-negative matrices can be traced back at least to those work on the convergence of non-homogeneous Markov chains [40]. Ever since then, various necessary and/or sufficient conditions for the convergence have been constructed, and several classical matrix theory books summarizing known results in this area have been published [44, 90, 42]. In the classical results, the set of stochastic, indecomposable, aperiodic matrices, called SIA matrices, has attracted a lot of attention, and the results on SIA matrices [101] have been used to prove that the agents can reach an agreement on the value of a variable of common interest using distributed nearest neighbor rules [47].

In many coordination algorithms, it is assumed that the agents in a network can update their states synchronously. However, in practice, the clock installed at each agent is often not synchronized with each other and the agents can only update according to their own clocks. Consequently, even when the synchronous coordination strategy converges, one still needs to check whether the same strategy implemented asynchronously still converges. To this end, in Chapter 2, we first reexamine a subclass of the SIA matices, the Sarymsakov class of stochastic matrices, which is introduced by Sarymsakov in [86] and redefined by Seneta in [89]. The Sarymsakov class is a semigroup under matrix multiplication and contains the set of scrambling matrices [89] as a subclass. We show that by generalizing the definition of the Sarymsakov class, we can make a connection to those much better understood SIA matrices. We further develop a new necessary and sufficient condition for the convergence of backward infinite products of stochastic matrices in terms of matrices from the Sarymsakov class of stochastic matrices in Chapter 3. Then we consider the coordination task for multi-agent systems when the agents update their states asynchronously. We prove that if the update matrix, when all the agents update synchronously, is a scrambling matrix, one can guarantee the system's convergence when the agents update asynchronously.

Much of the work on distributed algorithms has assumed that all the agents in a network are working cooperatively to reach an agreement, which implies that a neighboring pair of agents always contribute to decrease the relative difference as if they are attractive to each other. The cooperation between a pair of agents is modeled by the positive coupling between them, which corresponds to an edge with some positive weight in the interaction graph. However, typical coupled multi-agent systems indicate that conflict between pairs of agents in a network is ubiquitous. In neural networks, the coupling between neurons can be either excitatory or inhibitory [100]; in robotic teams, the interaction between self-interested robots can be either collaborative or competitive [13]; in social networks, the relationship between people can be either friendly or antagonistic [98].

In social network theory, a network that characterizes the friendly and antagonistic relationships among individuals is modeled by a signed graph. A signed graph is structurally balanced if it can be split into two factions, where each faction contains only friendly relationships while individuals from different factions are antagonistic [98]. In a static structurally balanced network, the states of the agents asymptotically converge to two opposite values, where the individuals in the same faction hold the same value, while the states of those from different factions are opposite [2, 3]. It is more challenging and interesting to investigate the dynamical behavior of the system under dynamically changing network topologies, where the networks may not be structurally balanced all the time or the bipartitions that divide the networks into two opposing factions may change with time.

This motivates us to study distributed algorithms in the presence of both positive and negative couplings. They generalize distributed algorithms in [47, 81, 16], where the graphs characterizing the interactions among the agents only consist of positive couplings and they are structurally balanced since one of the two factions is empty. When the network topology is static, the states of the agents polarize in a structurally balanced network, while in a structurally unbalanced network they converge to zero asymptotically. In the case when the network topologies are time-varying, polarization of the states of the agents will appear under some connectivity conditions if all the networks involved are structurally balanced and maintain a common bipartition of two opposing factions; otherwise, polarization of the states of the agents will not happen and instead, the states of all the agents asymptotically agree and converge to zero.

In the literature, various algorithms have been successfully constructed to cause all the agents in a group to converge to the same value asymptotically [47, 81, 16]. At the same time, there is an emerging trend to study how an interconnected group may incorporate or evolve into different sub-groups, called clusters [106]. In nature, multi-species foraging groups have been observed, such as flocks of bark foraging birds [32], in which birds have to coordinate through interactions with peers in their own and other species. In the study of social networks, the Krause model [43] describes how the agents with bounded confidence levels evolve into different clusters, where the agents in the same cluster hold the same opinion in the end. The clustering behavior is also potentially useful for the formation control problem for teams of autonomous agents [4]. In [4], one of the main research problems that have been surveyed is to split a formation into sub-formations in order to accomplish covering tasks or avoid obstacles.

Motivated by the reported clustering phenomena, in Chapter 5, we study the cluster synchronization problem, in which a coupled multi-agent system is required to split into several clusters, so that the agents synchronize with one another in the same cluster, but differences exist between different clusters. We are interested in identifying the approaches that might lead to clustering behavior in diffusively coupled networks that have mainly been used for synchronization study. We present three different approaches to realize clustering behavior in connected diffusively-coupled networks. When analyzing the three mechanisms, we also list related results that are scattered in the literature and make comparison when possible.

Furthermore, we make a connection to the controllability problem of multi-agent systems. A dynamical system is said to be *controllable* if under suitable control actions as the system's inputs, the system's state can be driven from any initial values to any desired final values within finite time [49]. For an interconnected multi-agent system, it is of great importance to know whether collective behavior can be achieved by controlling only a portion of the agents. This is fundamental to the design of effective distributed control algorithms. Tools from graph theory have been employed to attack this problem. Equitable partitions and almost equitable partitions are utilized to provide bounds for the controllable subspace of a multi-agent system [79, 63, 19, 113]. We generalize the notions to general directed weighted graphs and provide upper and lower bounds for the system's controllable subspace and show that those diffusively coupled multi-agent networks that are not controllable tend to realize cluster synchronization as time goes to infinity.

While physical devices, such as computational units, sensors and actuators, are more and more frequently working together over distances, people are more and more concerned with the problem of how to synchronize the clocks that are installed at those physical devices and connected through wired and/or wireless data networks [36]. The importance of clock synchronization can also be seen from examples of converging coordination algorithms that may not converge any more, when the clock installed at each agent is not synchronized with each other and the agents can only update according to their own clocks. Clock synchronization has been discussed intensively in the area of theoretical computer science especially in the 1980's [52, 92], and various impossibility results and bounds for synchronization errors have been reported [62, 61]. More recently, with the growing interest in the application of large-scale networks, in particular ad hoc and sensor networks, clock synchronization problems have attracted considerable attention [67, 8, 84, 76, 56, 57, 107].

Very recently, Freris, Graham and Kumar have shown that in an idealized setting the clocks *cannot* be synchronized precisely in distributed networks when asymmetric time delays are present [37]. This result is obtained by using tools from linear system theory and consistent with the results obtained previously in theoretical computer science. Such impossibility results point out insightfully the fundamental limit of distributed clock synchronization strategies and underscore the urgent need to carry out in-depth theoretical analysis for various clock synchronization protocols. On the other hand, in engineering practice when clocks are adjusted repeatedly to compensate the differences between their time displays, their displays can indeed get synchronized within an acceptable level of accuracy in a distributed fashion. In [94], the Time-Diffusion synchronization Protocol (TDP) has been proposed to enable sensor networks to synchronize their clocks with bounded errors. In [53], both synchronous and asynchronous versions of a rate-based diffusion protocol have been discussed, in which clocks adjust their displays repeatedly by taking the weighted average of the displays of themselves and their adjacent clocks. IEEE 1588 protocol [46] has been applied widely to networked measurement and control systems.

This motivates us to study the clock synchronization errors in the presence of asymmetric time delays in a network based on similar models for clocks as in [37]. By updating all clocks repeatedly, we are able to derive explicit expressions of the synchronization errors in steady states, which are within an acceptable range even when the time delays are asymmetric.

In short, the contributions of this thesis can be summarized as follows.

1. We reexamine the Sarymsakov class of stochastic matrices and make a connection to better understood SIA matrices.

2. We develop a new necessary and sufficient condition for the convergence of backward products of stochastic matrices and apply the results to derive new sufficient conditions that can guide the asynchronous implementation of coordination algorithms for multi-agent systems.

3. We develop sufficient conditions for the agents in a network to polarize or to reach an agreement for distributed algorithms in the presence of positive and

negative couplings under dynamically changing interaction topologies.

4. We propose three different distributed algorithms that may lead to clustering behavior of interacting agents in connected networks.

5. We provide an upper bound and a lower bound for the controllable subspace for a general diffusively coupled multi-agent system and show that those diffusively coupled multi-agent networks that are not controllable tend to realize cluster synchronization as time goes to infinity.

6. We determine clock synchronization errors in distributed networks in the presence of asymmetric time delays and show that the synchronization errors can be bounded within an acceptable level of accuracy that are determined by the degree of asymmetry in time delays.

1.3 Outline of this thesis

This thesis is structured as follows. Chapter 2 introduces basic notation and definitions that will be used throughout the thesis. It is followed by introductory terminologies and results in graph theory and algebraic graph theory, and basic results on nonnegative matrices and stochastic matrices. We also review some existing results on the convergence of distributed coordination algorithms.

In Chapter 3, some classical results on products of stochastic matrices are reviewed and a new necessary and sufficient condition is constructed by making use of the matrices in the Sarymsakov class. We then discuss a discrete-time averaging algorithm, which is implemented asynchronously to cope with the practical constraint that agents may not have access to a common clock. The set of scrambling stochastic matrices, a subclass of the Sarymsakov class, is utilized to establish the convergence of the agents' states based on the convergence results on products of stochastic matrices. The results presented in this chapter are extensions of those in [109].

In Chapter 4, we study distributed algorithms in the presence of positive and negative couplings. We discuss both cases when the network topologies are static and time-varying. In the case when the network topologies are time-varying, the states of the agents polarize under some connectivity conditions if all the networks involved are structurally balanced and maintain a common bipartition of two opposing factions. If structurally unbalanced networks arise often enough as time evolves, then the states of all the agents asymptotically agree and converge to zero.

Chapter 5 shows three different distributed algorithms that may lead to clustering behavior of coupled agents in connected networks. The first approach is that agents have different self-dynamics, and those agents having the same self-dynamics may evolve into the same cluster. When the agents' self-dynamics are identical, we present two other approaches by which cluster synchronization might be achieved. One is the presence of delays and the other is the existence of both positive and negative couplings between the agents. Some sufficient and/or necessary conditions are constructed to guarantee cluster synchronization. The results presented in this chapter have been published in [105, 106].

Chapter 6 discloses the relationship between the controllability problem and the cluster synchronization problem of complex multi-agent systems. We first define generalized equitable partitions and almost equitable partitions for general directed weighted graphs and then we are able to provide an upper bound and a lower bound for the controllable subspace for a general diffusively coupled multi-agent system. Furthermore, we show that those diffusively coupled multi-agent networks that are not controllable are in general easier to realize cluster synchronization. The results presented in this chapter are extensions of those in [108].

In Chapter 7, we determine clock synchronization errors in distributed networks in the presence of communication time delays. We show that the synchronization errors can be bounded within an acceptable level of accuracy that are determined by the degree of asymmetry in time delays. After studying the basic case of synchronizing two clocks in the two-way message passing process, we first analyze the directed ring networks, in which neighboring clocks are likely to experience severe asymmetric time delays. We then discuss connected undirected networks with two-way message passing between each pair of adjacent nodes. In the end, we expand the discussions to networks with directed topologies that are strongly connected. The results presented in this chapter are extensions of those in [107].

Concluding remarks and recommendations for future research are given in Chapter 8.

Chapter 2

Mathematical Preliminaries

In this chapter, we first introduce the general notation and definitions that will be used throughout the thesis. Some fundamental knowledge of graph theory and matrix theory is reviewed. This plays a crucial role in the convergence analysis of distributed coordination algorithms for multi-agent systems. The last section in this chapter reviews both discrete-time and continuous-time coordination algorithms that are well-studied in the past decade and collects some fundamental convergence results.

2.1 Basic notation and definitions

Let $\mathbb{R}$ denote the field of real numbers. Let N be a positive integer. $\mathbb{R}^N$ denotes the N-dimensional Euclidean space. I_N and $O_{M\times N}$ denote the $N \times N$ identity matrix and the $M \times N$ zero matrix, respectively. I and O denote the identity matrix and zero matrix with compatible dimension, respectively. $\mathbf{1}_N$ and $\mathbf{0}_N$ represent N-dimensional column vectors with all ones and all zeros, respectively. $\mathbf{1}$ denotes a column vector with all ones with compatible dimension. $\mathbf{0}$ denotes a column vector with all zeros with compatible dimension.

Let $A \in \mathbb{R}^{M\times N}$ be an $M \times N$ matrix. $\text{rank}(A)$ denotes the rank of A. $ker(A)$ denotes the kernel of A defined by $\{x \in \mathbb{R}^N | Ax = 0\}$ and $im(A)$ denotes the image of A defined by $\{y \in \mathbb{R}^M | y = Ax,\ \forall x \in \mathbb{R}^N\}$. We write $A \geqslant 0$ if $a_{ij} \geqslant 0,\ i = 1, \ldots, M,\ j = 1, \ldots, N$, and we say A is a nonnegative matrix. We write $A > 0$ if $a_{ij} > 0,\ i = 1, \ldots, M,\ j = 1, \ldots, N$, and we say A is a positive matrix. If $M = N$, A is a square matrix. A positive definite matrix A is denoted by $A \succ 0$ and a positive semi-definite A is denoted by $A \succcurlyeq 0$. Correspondingly, a negative definite matrix and a negative semi-definite matrix are denoted by $A \prec 0$ and $A \preccurlyeq 0$, respectively.

For two arbitrary matrices $A = (a_{ij})_{m\times n}$ and $B = (b_{ij})_{p\times q}$, the Kronecker product of A and B is defined by

$$A \otimes B = \begin{bmatrix} a_{11}B & a_{12}B & \cdots & a_{1n}B \\ a_{21}B & a_{22}B & \cdots & a_{2n}B \\ \vdots & \vdots & \ddots & \vdots \\ a_{m1}B & a_{m2}B & \cdots & a_{mn}B \end{bmatrix}.$$

For an $N \times N$ matrix A, assume that $\lambda_1(A),\ \lambda_2(A), \ldots, \lambda_N(A)$ are the eigenvalues of A and $Re(\lambda_i(A))$ denotes the real part of the eigenvalue $\lambda_i(A)$. $\rho(A)$ denotes the spectral radius of A, which is defined by $\rho(A) = \max_{1 \leqslant i \leqslant N}\{|\lambda_i(A)|\}$. A^s is defined by $A^s = \frac{1}{2}(A + A^T)$.

$dim(\cdot)$ denotes the dimension of a vector or a space. For a set S, let $|S|$ be the cardinality of S. Let $x = [x_1, \ldots, x_N]^T$ be an N-dimensional real vector. $||x||_2$ is the Euclidean norm of x defined by $||x||_2 = \sqrt{x^T x}$ and $||x||_\infty = \max\{|x_1|, \ldots, |x_N|\}$ is the max norm of x. $diag(x)$ denotes the diagonal matrix with the vector x on its diagonal.

2.2 Basics of graph theory

Graph theory serves as a fundamental and powerful tool in the study of network science. Graphs can be conveniently used to describe the topologies of networks and in later chapters we will frequently use them to visualize the interaction topologies among the agents or the communication topologies among the clocks in a network. This material can be found in many books on graph theory, for example [25, 30].

A graph consists of a vertex set and a set of edges connecting the vertices. Let a graph consisting of N vertices be denoted by $\mathbb{G} = (\mathcal{V}, \mathcal{E})$, with the vertex set $\mathcal{V} = \{v_1, v_2, \ldots, v_N\}$ and the edge set $\mathcal{E} \subseteq \mathcal{V} \times \mathcal{V}$. We will use node and vertex interchangeably in later chapters. Conventionally, when we utilize a graph $\mathbb{G}$ to represent the interaction topology among the agents, vertex v_i in the graph represents agent i in a multi-agent system. In an *undirected graph*, the edges in $\mathcal{E}$ are denoted by unordered pairs of vertices. $(v_i, v_j) \in \mathcal{E}$ if and only if there is an edge connecting v_i and v_j. In contrast, a *directed graph* is defined by an edge set consisting of ordered pairs of vertices; that is $(v_i, v_j) \in \mathcal{E}$ does not necessarily imply $(v_j, v_i) \in \mathcal{E}$ (see Fig. 2.1(a) for an example). For an edge $(v_i, v_j) \in \mathcal{E}$ in a directed graph, v_i is called the *parent vertex* and v_j is called the *child vertex*. If not explicitly stated, throughout this dissertation, we only consider graphs without self-loops, i.e. $(v_i, v_i) \notin \mathcal{E}$. An undirected graph can be viewed as a directed graph if every undirected edge (v_i, v_j) is represented by two directed edges (v_i, v_j) and (v_j, v_i). The *union* of a collection of graphs is a graph whose vertex and edge sets are the unions of the vertex and edge sets of the graphs in the collection.

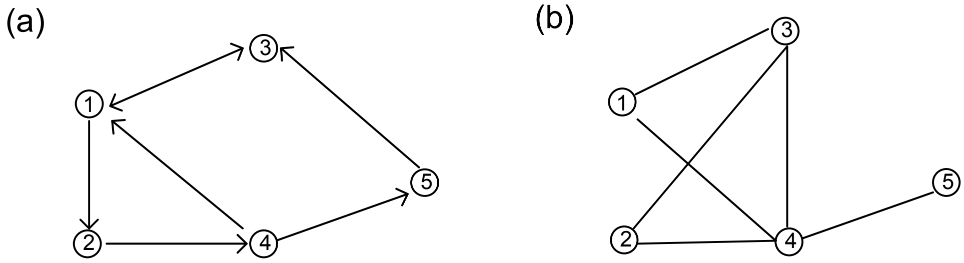


Figure 2.1: *(a) A directed strongly connected graph with five vertices; (b) an undirected connected graph with five vertices.*

In a directed graph $\mathbb{G}$, a *directed path* of length k is a sequence of distinct vertices $v_{i_1}, \ldots, v_{i_{k+1}}$ such that $(v_{i_s}, v_{i_{s+1}}) \in \mathcal{E}$ for $s = 1, \ldots, k$. An undirected path in an *undirected graph* is defined analogously. The *distance* from v_i to v_j is the length of the shortest path from v_i to v_j and is denoted by $dist(v_i, v_j)$. We define $dist(v_i, v_i) = 0$ for any v_i. Note that in directed graphs, $dist(v_i, v_j)$ is in general not equal to $dist(v_j, v_i)$. In a directed graph, a *cycle* is a directed path that starts and ends at the same vertex. A directed graph is said to be *strongly connected* if there is a directed path from every vertex to every other vertex. An undirected graph is connected if there is an undirected path from every vertex to every other vertex. The *diameter* of a graph $\mathbb{G}$ is defined by $diam(\mathbb{G}) = \max_{v_i, v_j \in \mathcal{V}} dist(v_i, v_j)$. Obviously, $1 \leqslant diam(\mathbb{G}) \leqslant N - 1$ when $\mathbb{G}$ is strongly connected and $N > 1$. A *directed tree* is a directed graph in which every vertex has only one parent except for one vertex, called the root, which has no parent and from which there is a directed path to every other vertex. For undirected graphs, a tree is a graph in which every pair of vertices is connected by exactly one undirected path.

A *subgraph* $\mathbb{G}_1 = (\mathcal{V}_1, \mathcal{E}_1)$ of $\mathbb{G} = (\mathcal{V}, \mathcal{E})$ is a graph such that $\mathcal{V}_1 \subseteq \mathcal{V}$ and $\mathcal{E}_1 \subseteq \mathcal{E} \cap (\mathcal{V}_1 \times \mathcal{V}_1)$. A *directed spanning tree* $(\mathcal{V}, \mathcal{E}_1)$ of a directed graph $(\mathcal{V}, \mathcal{E})$ is a subgraph of $(\mathcal{V}, \mathcal{E})$ such that $(\mathcal{V}, \mathcal{E}_1)$ is a directed tree. An *undirected spanning tree* of an undirected graph is defined analogously. We say a directed graph contains a directed spanning tree if a directed spanning tree is a subgraph of this graph. It is noted that a directed graph contains a directed spanning tree if and only if it contains at least one vertex from which there is a directed path to every other vertex. This is a condition which is weaker than that a graph is strongly connected. In contrast, an undirected graph contains a spanning tree if and only if it is connected.

The *adjacency matrix* $A = (a_{ij})_{N \times N}$ of a directed graph $\mathbb{G} = (\mathcal{V}, \mathcal{E})$ is defined such that a_{ij} is nonzero if $(j,i) \in \mathcal{E}$ and $a_{ij} = 0$ if $(j,i) \notin \mathcal{E}$. It is noted that $a_{ii} = 0$, since there is no self-loop in a graph. The *in-degree* of vertex i is defined as $d_i^{in} = \sum_{j=1,\ j\neq i}^{N} a_{ij}$ and the *out-degree* of vertex i is defined as $d_i^{out} = \sum_{j=1,\ j\neq i}^{N} a_{ji}$. The *in-degree* and *out-degree matrices* of a graph are defined as $D^{in} = diag\{d_1^{in}, \ldots, d_N^{in}\}$ and $D^{out} = diag\{d_1^{out}, \ldots, d_N^{out}\}$. A graph is called *balanced* if $d_i^{in} = d_i^{out}$ for all $i = 1, \ldots, N$. For any undirected graph, the adjacency matrix A is symmetric, and thus it is balanced. If the weights are irrelevant, then a_{ij} is equal to 1 for all $(j,i) \in \mathcal{E}$. In this case, the in-degree and out-degree of each vertex reduce to the number of edges pointing to and pointing out from this vertex, respectively.

The Laplacian matrix of a directed graph is defined as $L = D^{in} - A$; that is

$$l_{ii} = \sum_{j=1,\ j\neq i}^{N} a_{ij}, \quad l_{ij} = -a_{ij},\ i \neq j. \tag{2.1}$$

The spectral properties of the Laplacian matrix play an important role in the convergence study of distributed coordination algorithms. Here we introduce some useful properties on the spectrum of the Laplacian matrix.

The following result, often called the *Gersgorin disc theorem*, reveals that the eigenvalues of a matrix lie in some easily computed discs centered at the diagonal elements of the matrix.

2.2.1. LEMMA. *[44] Let $A = (a_{ij})_{N\times N}$ and let $R_i(A) = \sum_{j=1,j\neq i}^{N} |a_{ij}|,\ 1 \leqslant i \leqslant N$ denote the deleted absolute row sums of A. Then all the eigenvalues of A are located in the union of N discs*

$$\bigcup_{i=1}^{N} \left\{ z \in \mathbb{C} : |z - a_{ii}| \leqslant \sum_{j=1,j\neq i}^{N} |a_{ij}| = R_i(A) \right\} = G(A). \tag{2.2}$$

The region $G(A)$ in (2.2) is often called the *Gersgorin region* (for rows) of A; the individual discs in $G(A)$ are called *Gersgorin discs*, and the boundaries of these discs are called *Gersgorin circles*.

Assume that the weights of all the edges in a directed graph are positive; that is $a_{ij} > 0$ for all $(j,i) \in \mathcal{E}$. Then $l_{ij} = -a_{ij} \leqslant 0$, for $i \neq j$, and $\sum_{j=1}^{N} l_{ij} = d_i^{in} - \sum_{j=1,\ j\neq i}^{N} a_{ij} = 0$, for all $i = 1, \ldots, N$. It follows that L has zero row sums and 0 is an eigenvalue of L with an eigenvector $\mathbf{1}_N = [1, \ldots, 1]^T$. Note that L is diagonally dominant and it has nonnegative diagonal elements. Applying Lemma 2.2.1, it is clear that the nonzero eigenvalues of L all have positive real parts.

2.2.2. LEMMA. *[81] Assume that the adjacency matrix A are nonnegative, that is $a_{ij} \geqslant 0,\ i,j = 1, \ldots, N$. 0 is a simple eigenvalue of the Laplacian matrix L of a directed graph if and only if the graph contains a directed spanning tree. In addition, all the other eigenvalues of L have positive real parts.*

For an undirected graph, the adjacency matrix A is symmetric and so is L. Let $\lambda_i(L)$ be the eigenvalues of L and assume that they are arranged in an increasing order $\lambda_1(L) \leqslant \lambda_2(L) \cdots \leqslant \lambda_N(L)$. Then $\lambda_1(L) = 0$ and $\lambda_2(L)$ is called the *algebraic connectivity* of an undirected graph [65], which is strictly positive if and only if the graph is connected.

If the weights of the edges in a graph we concern can take both positive and negative values, then the graph becomes a *signed graph*. The adjacency matrix hence may contain negative entries and is thus called a *signed adjacency matrix*. The definition of the Laplacian matrix in (2.1) will render a matrix that still has an eigenvalue 0 but may have eigenvalues with negative real parts. To overcome this, we introduce the extended definition of the Laplacian matrix for a signed graph, and called it the *signed Laplacian matrix* [45]. The signed Laplacian matrix is given by

$$l_{ii} = \sum_{j=1,\ j\neq i}^{N} |a_{ij}|, \quad l_{ij} = -a_{ij},\ i \neq j. \tag{2.3}$$

It is easy to see that the signed Laplacian matrix is diagonally dominant and thus its nonzero eigenvalues all have positive real parts.

Given a matrix $G = (g_{ij})_{N\times N}$, a directed weighted graph $\mathbb{G}(G)$ associated to G can be defined to visualize the nonzero entries of G, which will be used extensively in later chapters to describe the couplings among the agents in a network. For the sake of convenience, we have slightly modified Definition 6.2.11 in [44] to get the following one.

2.2.1. Definition. *Given a matrix $G = (g_{ij})_{N\times N}$, the directed weighted graph associated to G, denoted by $\mathbb{G}(G) = (\mathcal{V}, \mathcal{E})$, is a directed graph with the vertex set $\mathcal{V} = \{v_1, \ldots, v_N\}$ such that (v_i, v_j) is an edge of $\mathbb{G}(G)$ if and only if $i \neq j$ and $g_{ji} \neq 0$, and the weight associated with (v_i, v_j) is g_{ji}.*

Note that $\mathbb{G}(G)$ contains no self-loops, i.e., $(v_i, v_i) \notin \mathcal{E}$. There is one exception in Section 3.2, where self-loops are allowed in the associated graph of a matrix.

2.3 Basics of matrix theory

The class of nonnegative matrices, especially its subclass of stochastic matrices, plays an essential role in establishing the effectiveness of the distributed coordination algorithms. Next we introduce several well-studied classes of matrices and a newly defined class of matrices that will be used in the study of products of stochastic matrices and an asynchronous implementation of a distributed coordination algorithm in Chapter 3.

2.3.1. DEFINITION. *[44] A matrix $P = \{p_{ij}\}_{N\times N}$ is said to be reducible if either*
(a) $N = 1$ and $P = 0$; or
(b) $N \geqslant 2$, there is a permutation matrix U and there is some integer r with $1 \leqslant r \leqslant N-1$ such that

$$U^T PU = \begin{bmatrix} B & C \\ O & D \end{bmatrix},$$

where $B \in \mathbb{R}^{r\times r}$, $C \in \mathbb{R}^{r\times N-r}$, $D \in \mathbb{R}^{N-r\times N-r}$, and O is the zero matrix. P is said to be irreducible if it is not reducible.

2.3.2. DEFINITION. *[44] A square matrix $P = \{p_{ij}\}_{N\times N}$ is called stochastic if it is nonnegative and $\sum_{j=1}^N p_{ij} = 1$ for all $i = 1, \ldots, N$.*

Consider a stochastic matrix P. For a set $\mathcal{A} \subseteq \{1, \ldots, N\}$, let $F_P(\mathcal{A})$ be the set of *one-stage consequent indices* [89] of $\mathcal{A}$ defined by $F_P(\mathcal{A}) = \{j : \ p_{ij} > 0 \text{ for some } i \in \mathcal{A}\}$.

2.3.3. DEFINITION. *[101] A stochastic matrix $P = \{p_{ij}\}_{N\times N}$ is indecomposable and aperiodic and thus called an SIA matrix if $\lim_{m\to\infty} P^m = \mathbf{1}c^T$, where $c = [c_1, \ldots, c_N]^T$ is some column vector satisfying $c_i \geqslant 0$ and $\sum_{i=1}^N c_i = 1$.*

2.3.4. DEFINITION. *[90] A square matrix $P = \{p_{ij}\}_{N\times N}$ belongs to the Sarymsakov class $\mathcal{K}$ if for any two disjoint nonempty subsets $\mathcal{A}$, $\tilde{\mathcal{A}} \subseteq \{1, \ldots, N\}$, either*

$$F_P(\mathcal{A}) \cap F_P(\tilde{\mathcal{A}}) \neq \varnothing \tag{2.4}$$

or

$$F_P(\mathcal{A}) \cap F_P(\tilde{\mathcal{A}}) = \varnothing \text{ and } |F_P(\mathcal{A}) \cup F_P(\tilde{\mathcal{A}})| > |\mathcal{A} \cup \tilde{\mathcal{A}}|. \tag{2.5}$$

2.3.5. DEFINITION. *[90] A square matrix P is called scrambling if for any pair of distinct row indices i and j, there always exists a column index k such that both p_{ik} and p_{jk} are positive.*

Obviously, from the definitions, a scrambling matrix belongs to the Sarymsakov class $\mathcal{K}$. It has been shown that any product of $N-1$ matrices from $\mathcal{K}$ is scrambling and a stochastic scrambling matrix is SIA [89]. Hence, any stochastic matrix belonging to the Sarymsakov class $\mathcal{K}$ must be an SIA matrix.

2.3.1. EXAMPLE. Let

$$P_1 = \begin{bmatrix} 1/2 & 0 & 1/2 \\ 1/2 & 1/2 & 0 \\ 0 & 1/2 & 1/2 \end{bmatrix}, \ P_2 = \begin{bmatrix} 1 & 0 & 0 \\ 1/2 & 1/2 & 0 \\ 0 & 1/2 & 1/2 \end{bmatrix}.$$

One can check that P_1 is a stochastic scrambling matrix and P_2 belongs to the Sarymsakov class $\mathcal{K}$. P_2 is not a scrambling matrix, since there is no column index k such that $(P_2)_{ik}$ and $(P_2)_{jk}$ are both positive for $i = 1$ and $j = 3$. Furthermore,

$$\lim_{m\to\infty} P_1^m = \frac{1}{3}\begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{bmatrix}, \quad \lim_{m\to\infty} P_2^m = \begin{bmatrix} 1 & 0 & 0 \\ 1 & 0 & 0 \\ 1 & 0 & 0 \end{bmatrix}.$$

Hence P_1 and P_2 are both SIA matrices. □

2.3.6. Definition. *[90] Let $P = \{p_{ij}\}_{N\times N}$ be an arbitrary stochastic matrix. $\tau(P) = \frac{1}{2}\max_{l,k}\sum_{s=1}^{N}|p_{ls} - p_{ks}| = 1 - \min_{l,k}\sum_{s=1}^{N}\min\{p_{ls}, p_{ks}\}$ is the coefficient of ergodicity of P.*

The following property of the coefficient of ergodicity is clear from its definition.

2.3.1. Lemma. *[90] Let $\tau(P)$ be the coefficient of ergodicity of a stochastic matrix P.*
(a) $0 \leqslant \tau(P) \leqslant 1$;
(b) $\tau(P) = 0$ if and only if $P = \mathbf{1}c^T$ for some vector c satisfying $c \geqslant 0,\ c^T\mathbf{1} = 1$;
(c) $\tau(P) < 1$ if and only if P is a scrambling matrix.

In fact, the set of all the stochastic matrices in Sarymsakov class is the largest known set of stochastic matrices, which is closed under matrix multiplication and the products of whose elements under mild conditions always converge. However, the definition of the Sarymsakov class is a bit obscure and thus it might seem difficult to place such matrices in relationship with some other categories of well-known stochastic matrices. To deal with this challenge, we take a closer look at the definition of the Sarymsakov class and explain its relationship to the SIA matrices.

From the definition of the Sarymsakov class $\mathcal{K}$, to verify whether a matrix belongs to $\mathcal{K}$ or not, one needs to check the set of one-stage consequent indices of any two disjoint sets of its indices. Said differently, the definition of the Sarymsakov class is tightly built upon the notion of one-stage consequent indices. Motivated by this, we try to explore what happens when we further look at the set of "k-stage consequent indices" of any two nonempty sets of its indices. Naturally, we will obtain a larger matrix set, which contains the Sarymsakov class $\mathcal{K}$. But surprisingly, as we will prove later, this larger set is exactly the set of SIA matrices.

For a stochastic matrix P and a set $\mathcal{A} \subseteq \{1, \ldots, N\}$, let $F_P^k(\mathcal{A})$ be the set of *k-stage consequent indices* of any nonempty set $\mathcal{A} \subseteq \{1, \ldots, N\}$, which is defined by

$$F_P^1(\mathcal{A}) = F_P(\mathcal{A}) \text{ and } F_P^k(\mathcal{A}) = F_P(F_P^{k-1}(\mathcal{A})) \text{ for } k \geqslant 2.$$

2.3.7. Definition. *A square matrix $P = \{p_{ij}\}_{N\times N}$ belongs to the class $\mathcal{W}$ if for any two disjoint nonempty subsets $\mathcal{A},\ \tilde{\mathcal{A}} \subseteq \{1,\ldots,N\}$, there exists an integer k such that either*

$$F_P^k(\mathcal{A}) \cap F_P^k(\tilde{\mathcal{A}}) \neq \varnothing, \tag{2.6}$$

or

$$F_P^k(\mathcal{A}) \cap F_P^k(\tilde{\mathcal{A}}) = \varnothing \text{ and } |F_P^k(\mathcal{A}) \cup F_P^k(\tilde{\mathcal{A}})| > |\mathcal{A} \cup \tilde{\mathcal{A}}|. \tag{2.7}$$

It is easy to see that $\mathcal{K} \subseteq \mathcal{W}$ since $k = 1$ in the definition of $\mathcal{K}$. The relationship between the newly defined class $\mathcal{W}$ and the class of SIA matrices can be summarized as following.

2.3.1. Theorem. *A stochastic matrix P is in $\mathcal{W}$ if and only if P is SIA.*

The proof of Theorem 2.3.1 makes use of the following Lemma.

2.3.2. Lemma. *[42] Let P be an $N \times N$ stochastic matrix. Then $F_P^k(\mathcal{A}) = F_{P^k}(\mathcal{A})$ for all subsets $\mathcal{A} \subseteq \{1,\ldots,N\}$.*

Proof of Theorem 2.3.1: (Sufficiency) Since P is SIA, there exists a positive integer k such that P^k has a column with only positive elements. From Lemma 2.3.2, one has

$$F_P^k(\mathcal{A}) \cap F_P^k(\tilde{\mathcal{A}}) = F_{P^k}(\mathcal{A}) \cap F_{P^k}(\tilde{\mathcal{A}}) \neq \varnothing$$

for any two disjoint nonempty subsets $\mathcal{A},\ \tilde{\mathcal{A}} \subseteq \{1,\ldots,N\}$. Thus $P \in \mathcal{W}$.
(Necessity) Let the number of all possible pairs of disjoint nonempty subsets be m and $k_1,\ldots,k_m$ be the corresponding positive integers such that either (2.6) or (2.7) holds. Let $s = \max\{k_1,\ldots,k_m\}$ and $l = (N-1)s$. We claim that for any two disjoint nonempty subsets $\mathcal{A}_1,\ \tilde{\mathcal{A}}_1 \subseteq \{1,\ldots,N\}$, $F_P^l(\mathcal{A}_1) \cap F_P^l(\tilde{\mathcal{A}}_1) \neq \varnothing$.

If this is not true, one has $F_P^i(\mathcal{A}_1) \cap F_P^i(\tilde{\mathcal{A}}_1) = \varnothing$, for all $i = 1,\ldots,l$. Since $P \in \mathcal{W}$, for $\mathcal{A}_1$ and $\tilde{\mathcal{A}}_1$, there exists a positive integer, without loss of generality, say k_1, such that

$$F_P^{k_1}(\mathcal{A}_1) \cap F_P^{k_1}(\tilde{\mathcal{A}}_1) = \varnothing \text{ and } |F_P^{k_1}(\mathcal{A}_1) \cup F_P^{k_1}(\tilde{\mathcal{A}}_1)| > |\mathcal{A}_1 \cup \tilde{\mathcal{A}}_1|.$$

Let $F_P^{k_1}(\mathcal{A}_1) = \mathcal{A}_2$ and $F_P^{k_1}(\tilde{\mathcal{A}}_1) = \tilde{\mathcal{A}}_2$. Then there exists a positive integer k_2 such that

$$F_P^{k_2}(\mathcal{A}_2) \cap F_P^{k_2}(\tilde{\mathcal{A}}_2) = \varnothing \text{ and } |F_P^{k_2}(A_2) \cup F_P^{k_2}(\tilde{\mathcal{A}}_2)| > |A_2 \cup \tilde{\mathcal{A}}_2| > |\mathcal{A}_1 \cup \tilde{\mathcal{A}}_1|.$$

Thus we can find a sequence of subsets $\mathcal{A}_2,\ \tilde{\mathcal{A}}_2,\ldots,\mathcal{A}_{N-1},\ \tilde{\mathcal{A}}_{N-1}$ and a sequence of positive integers $k_1,\ k_2,\ldots,k_{N-1}$, such that

$$F_P^{k_i}(\mathcal{A}_i) = \mathcal{A}_{i+1} \text{ and } F_P^{k_i}(\tilde{\mathcal{A}}_i) = \tilde{\mathcal{A}}_{i+1}$$

for $i = 1, \ldots, N-2$, and

$$F_P^{k_{N-1}}(\mathcal{A}_{N-1}) \cap F_P^{k_{N-1}}(\tilde{\mathcal{A}}_{N-1}) = \varnothing$$

and

$$|F_P^{k_{N-1}}(\mathcal{A}_{N-1}) \cup F_P^{k_{N-1}}(\tilde{\mathcal{A}}_{N-1})| > |\mathcal{A}_{N-1} \cup \tilde{\mathcal{A}}_{N-1}| > \cdots > |A_2 \cup \tilde{A}_2| > |A_1 \cup \tilde{A}_1| \geqslant 2.$$

It follows that

$$|F_P^{k_{N-1}}(\mathcal{A}_{N-1}) \cup F_P^{k_{N-1}}(\tilde{\mathcal{A}}_{N-1})| > N,$$

which contradicts the fact that the dimension of P is N.

From Lemma 2.3.2, one has that for any two disjoint nonempty subsets $\mathcal{A}_1,\ \tilde{\mathcal{A}}_1 \subseteq \{1, \ldots, N\}$,

$$F_{P^l}(\mathcal{A}_1) \cap F_{P^l}(\tilde{\mathcal{A}}_1) = F_P^l(\mathcal{A}_1) \cap F_P^l(\tilde{\mathcal{A}}_1) \neq \varnothing.$$

So $P^l \in \mathcal{K}$, which implies that P is SIA. □

2.3.2. Example. Let

$$P = \begin{bmatrix} 1 & 0 & 0 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \end{bmatrix}.$$

P is a stochastic matrix but P does not belong to the Sarymsakov class $\mathcal{K}$. To see this, take $\mathcal{A} = \{1\}$ and $\tilde{\mathcal{A}} = \{3\}$. One has that $F_P(\mathcal{A}) \cap F_P(\tilde{\mathcal{A}}) = \varnothing$ and $|F_P(\mathcal{A}) \cup F_P(\tilde{\mathcal{A}})| = |\mathcal{A} \cup \tilde{\mathcal{A}}| = 2$. If we take $k = 2$, then we have $F_P^k(\mathcal{A}) \cap F_P^k(\tilde{\mathcal{A}}) = \{1\} \neq \varnothing$. One can verify that P belongs to the class $\mathcal{W}$. Furthermore,

$$\lim_{m\to\infty} P^m = \begin{bmatrix} 1 & 0 & 0 \\ 1 & 0 & 0 \\ 1 & 0 & 0 \end{bmatrix}.$$

This verifies the effectiveness of Theorem 2.3.1. □

2.4 Distributed coordination algorithms

In this section, we review some well-studied distributed algorithms for multi-agent systems with discrete-time and continuous-time dynamics. These algorithms are often used to coordinate a group of agents to reach a consensus. By consensus we mean that all the agents in a network achieve a common value on some variable of interest asymptotically. A more rigorous definition will be given later. We are interested in systems that are described by first-order dynamics. The systems with second-order dynamics and general linear dynamics are not reviewed here, the interested reader is referred to [80, 87]. Various variations of these models have been developed to deal with practical constraints, for example, communication time-delays [17, 15, 110], asynchronous update events [17, 15, 110, 109], quantized measurements [50, 23, 21], and so on [80, 82]. There are also some other important issues that concern the convergence speed [74], distributed formation control [71], and so on.

For the discrete-time case, distributed averaging rules are often used to cause all the agents to reach an agreement. To be more specific, each agent updates its state to the weighted average of the states of all the other agents. Consider a system consisting of N agents, labeled by $1, \ldots, N$. The dynamics of the state of agent i can be described by

$$x_i(t+1) = \sum_{j=1}^{N} p_{ij}(t) x_j(t), \qquad t = 0, 1, \ldots. \tag{2.8}$$

where $x_i(t) \in \mathbb{R}$, and $p_{ij}(t) \geqslant 0$ is the nonnegative weight agent i assigned to agent j at time t when agent i updates its state. Let $x(t) = [x_1(t), \ldots, x_N(t)]^T \in \mathbb{R}^N$ and $P = (p_{ij}(t))_{N\times N} \geqslant 0$. We can write the system in a compact form

$$x(t+1) = P(t)x(t), \qquad t = 0, 1, \ldots. \tag{2.9}$$

The weights $p_{ij}(t)$ satisfy the following condition

$$\sum_{j=1}^{N} p_{ij}(t) = 1,\ p_{ii}(t) > 0,\ i = 1, \ldots, N. \tag{2.10}$$

From the condition (2.10), the row sums of $P(t)$ are all one and thus $P(t)$ is a stochastic matrix. When $P(t)$ is time-invariant, simply denote it as P. *Consensus is reached asymptotically* if for all initial values and all $i, j = 1, \ldots, N$, $x_i(t) - x_j(t) \to 0$, as $t \to \infty$.

2.4.1. LEMMA. *[80] Let A be a stochastic matrix. 1 is an eigenvalue of A and $\rho(A) = 1$. 1 is a simple eigenvalue of A if and only if its associated graph $\mathbb{G}(A)$ defined in Definition 2.2.1 has a directed spanning tree. Furthermore, if $\mathbb{G}(A)$ has a directed spanning tree and $a_{ii} > 0$, $i = 1, \ldots, N$, then 1 is the unique eigenvalue of maximum modulus.*

The following theorems establish the convergence results for discrete-time algorithm (2.9) under time-invariant and time-varying interaction topologies.

2.4.1. THEOREM. *[80] Suppose that $P(t) = P$ for all $t = 0, 1, \ldots$. The discrete-time system (2.9) with time-invariant interaction topology achieves consensus asymptotically if and only if the directed graph $\mathbb{G}(P)$ has a directed spanning tree. In particular, $x_i(t) \to \sum_{i=1}^{N} v_i x_i(0)$, where $v = [v_1, \ldots, v_N]^T \geqslant 0$ satisfies $v^T P = v$ and $\mathbf{1}^T v = 1$.*

2.4.2. Theorem. *[80] Suppose that the nonzero entries of the stochastic matrices $P(t)$ in system (2.9) are uniformly lower bounded, that is $p_{ij}(t) \geqslant \gamma$, for all $(j,i) \in \mathcal{E}(t)$ and all t, where $0 < \gamma < 1$. The discrete-time system (2.9) achieves consensus asymptotically if there exists an infinite sequence of contiguous, nonempty, uniformly bounded time intervals $[t_k, t_{k+1})$, $k = 0, 1, 2, \ldots$, starting at $t_0 = 0$, with the property that the union of the directed graphs across each interval has a directed spanning tree.*

For the continuous-time case, the consensus algorithm is given by

$$\dot{x}_i = -\sum_{j=1}^{N} a_{ij}(t)(x_i - x_j), \qquad i = 1, \ldots, N, \tag{2.11}$$

where $a_{ij}(t)$ is the ijth entry of the adjacency matrix $A(t)$ of the interaction graph $\mathbb{G}(t)$ at time t. Note that $a_{ij}(t) > 0$ if $(j,i) \in \mathcal{E}(t)$ and $a_{ij}(t) = 0$ otherwise, $\forall j \neq i$. Let $L(t)$ be the Laplacian matrix at time t and let $x(t) = [x_1(t), \ldots, x_N(t)]^T$. System (2.11) can be written in a compact form

$$\dot{x} = -L(t)x. \tag{2.12}$$

The definition $l_{ii} = -\sum_{j=1, j\neq i}^{N} l_{ij}$ guarantees that the inter-agent couplings are d-iffusive, and hence such networks are also called *diffusively coupled networks*. *Consensus is reached asymptotically* if for all initial values and all $i, j = 1, \ldots, N$, $x_i(t) - x_j(t) \to 0$, as $t \to \infty$.

The following theorem gives a necessary and sufficient condition for consensus with a time-invariant interaction topology and constant a_{ij}.

2.4.3. Theorem. *[80] Suppose that $A(t) = A$ is constant for all $t \geqslant 0$. The continuous-time system (2.12) achieves consensus asymptotically if and only if the directed graph $\mathbb{G}(A)$ has a directed spanning tree. In particular, $x_i(t) \to \sum_{i=1}^{N} v_i x_i(0)$, as $t \to \infty$, where $v = [v_1, \ldots, v_N]^T \geqslant 0$ satisfies $v^T L = 0$ and $\mathbf{1}^T v = 1$.*

When algorithm (2.12) is carried out under dynamically changing interaction topologies, we assume that $A(t)$ and $L(t)$ are both piecewise continuous.

2.4.4. Theorem. *[80] Suppose that $A(t)$ is piecewise continuous and its positive entries are uniformly lower and upper bounded, that is $a_{ij}(t) \in [\underline{a}, \bar{a}]$, for all t and $a_{ij}(t) \neq 0$, where $0 < \underline{a} < \bar{a}$. Let $t_0, t_1, \ldots,$ be the time sequence corresponding to the times at which $\mathbb{G}(t)$ switches, where it is assumed that $t_{k+1} - t_k \geqslant t_L$, $\forall k = 1, 2, \ldots,$ with t_L a positive constant. The continuous-time system (2.12) achieves consensus asymptotically if there exists an infinite sequence of contiguous, nonempty, uniformly bounded time intervals $[t_{i_k}, t_{i_{k+1}})$, $k = 0, 1, 2, \ldots$, starting at $t_{i_0} = 0$, with the property that the union of the directed graphs across each interval has a directed spanning tree.*

The discrete-time and continuous-time algorithms introduced here include several models like the Vicsek model [47, 97] as special cases, for more details please refer to [80, 72]. Note that in systems (2.8) and (2.11), the weights $p_{ij}(t)$ and $a_{ij}(t)$ are assumed to be nonnegative. This means that all the agents in a network always contribute to decrease the relative difference in order to reach an agreement. As pointed out in Chapter 1, it is meaningful to take negative couplings into account, which characterize competitive or antagonistic relationships between the agents. In Chapter 4, we generalize the discussions on systems (2.8) and (2.11) to the setting when the weights $p_{ij}(t)$ and $a_{ij}(t)$ can be positive and negative. Instead of reaching a consensus, polarization of the states of the agents arises in a structurally balanced network that can be split into two antagonistic factions. In Chapter 5, algorithm (2.11) is used to generate clustering behavior in a network by incorporating negative couplings.

Chapter 3

Asynchronous implementation of a distributed coordination algorithm

There are a number of results discussing how to use distributed averaging rules to cause a group of agents to reach an agreement on the value of a variable of interest as shown in the previous chapter. In most of the established results, the agents are assumed to update their states synchronously. In practice, however, the agents may not have access to a common clock and only update according to their own clocks. Consequently, even when the synchronized coordination algorithm converges, one still needs to check whether the same strategy implemented asynchronously converges. In this chapter, we focus on an asynchronous implementation of a distributed coordination algorithm. Some classic results on the convergence of products of stochastic matrices are reviewed and a new necessary and sufficient condition is constructed by making use of the matrices in the Sarymsakov class. The set of scrambling stochastic matrices, a subclass of the Sarymsakov class, is utilized to establish the convergence of the agents' states when there is no common clock for the agents to synchronize their update actions.

3.1 Problem formulation

Consider a system consisting of N agents, labeled by $1, \ldots, N$. All agents are required to reach an agreement on the value of a variable of interest and there have been rich results on how to achieve this goal. To be more specific, we denote the state of agent i, $1 \leqslant i \leqslant N$, by $x_i \in \mathbb{R}$. As in (2.8) with a time-invariant update matrix, agent i's state updates according to

$$x_i(t+1) = \sum_{j=1}^{N} p_{ij} x_j(t),\ t = 0, 1, \ldots, \tag{3.1}$$

where $p_{ij} \geqslant 0$ and $\sum_{j=1}^{N} p_{ij} = 1$. Hence, when all the agents' update actions are perfectly synchronized according to a common clock, the N-agent system's dynamics can be described by

$$x(t+1) = Px(t),\ t = 0, 1, \ldots, \tag{3.2}$$

where P is the $N \times N$ stochastic matrix whose ijth element is p_{ij}, and $x(t) = [x_1(t), \ldots, x_N(t)]^T$ is the state of the system. Then it has been proved that when the matrix P is SIA, the states x_i of all the agents converge to a common value asymptotically exponentially fast [81, 47].

In practice, the agents may not have access to a common clock and only update according to their own clocks. Consequently, it is important to check whether the converging coordination strategy (3.2) implemented asynchronously still converges. Note that the analysis of the convergence of asynchronous algorithms is usually challenging, and it is in general difficult to define the state of the asynchronous system even when its synchronous counterpart is well defined [10, 15].

Now we consider a possible asynchronous implementation of the distributed averaging algorithm described by (3.2). We allow each agent to update its state independently at times determined by its own clock. We will give a sufficient condition on the matrix P to ensure the convergence of the asynchronous implementation of the update scheme.

We assume that the agents' clocks can be described by linear models and have the same skew but different offsets [37, 107]. We first ignore the case that two or more agents update exactly at the same time. So one can carry out the procedure of *analytic synchronization*, at the end of which we obtain the set $\mathcal{T} = \{0, 1, 2, \ldots\}$ by relabeling all the agents' update event times. For more detailed description of analytic synchronization, the interested reader is referred to [10, 54, 15]. In the following, we consider the system that evolves according to the time sequence $\mathcal{T}$. If each agent chooses to update periodically, then the overall system becomes periodic as well with period N.

Now consider at time t, agent $c_t,\ 1 \leqslant c_t \leqslant N$ updates. Then its state satisfies

$$x_{c_t}(t+1) = \sum_{j=1}^{N} p_{c_t j} x_j(t),\ t = 0, 1, 2, \ldots, \tag{3.3}$$

and correspondingly for all the other agents, we have

$$x_j(t+1) = x_j(t),\ j \neq c_t,\ j = 1, \ldots, N,\ t = 0, 1, 2, \ldots. \tag{3.4}$$

Then again with the definition of the system's state to be $x(t) = [x_1(t), \ldots, x_N(t)]^T$, one can rewrite (3.3) and (3.4) in a compact form

$$x(t+1) = P_{c_t} x(t),\ t = 0, 1, 2, \ldots, \tag{3.5}$$

where

$$P_{c_t} = \begin{bmatrix} 1 & \cdots & 0 & 0 & \cdots & 0 \\ \vdots & \ddots & \vdots & \vdots & \cdots & \vdots \\ 0 & \cdots & 1 & 0 & \cdots & 0 \\ p_{c_t 1} & \cdots & p_{c_t, c_t - 1} & p_{c_t c_t} & \cdots & p_{c_t N} \\ \vdots & \cdots & \cdots & \vdots & \ddots & \vdots \\ 0 & \cdots & 0 & 0 & 0 & 1 \end{bmatrix}. \tag{3.6}$$

The solution to the system (3.5) can be written as

$$x(t+1) = P_{c_t} P_{c_{t-1}} \cdots P_{c_1} P_{c_0} x(0). \tag{3.7}$$

The asymptotic behavior of the solution depends on the property of the infinite backward product $\cdots P_{c_t} \cdots P_{c_1} P_{c_0}$. Each P_{c_t} in the product is a stochastic matrix. The convergence study of products of stochastic matrices has proven to be crucial in establishing the effectiveness of distributed coordination algorithms for multi-agent systems. In the next section we review some classic and recent results on products of stochastic matrices and then use them to establish a sufficient condition on P to ensure the convergence of asynchronous system (3.5).

3.2 Products of stochastic matrices

Consider a compact set $\mathcal{P}$ of $N \times N$ stochastic matrices. The following conditions have been constructed in [101] and [6] to guarantee the convergence of the backward product of matrices from $\mathcal{P}$, which involve SIA matrices and scrambling matrices defined in Definitions 2.3.3 and 2.3.5.

C1. For each integer $k \geqslant 1$ and any $P(i) \in \mathcal{P},\ 1 \leqslant i \leqslant k$, the stochastic matrix $P(k) \cdots P(1)$ is SIA.

C2. There is an integer $\nu \geqslant 1$ such that for each $k \geqslant \nu$ and any $P(i) \in \mathcal{P},\ 1 \leqslant i \leqslant k$, the matrix $P(k) \cdots P(1)$ is scrambling.

C3. There is an integer $\mu \geqslant 1$ such that for each $k \geqslant \mu$ and any $P(i) \in \mathcal{P},\ 1 \leqslant i \leqslant k$, the matrix $P(k) \cdots P(1)$ has a column with only positive elements.

In [6], the relationships between these three conditions have been discussed.

3.2.1. Proposition. [6] *Conditions C1, C2 and C3 are equivalent.*

Then these three conditions are shown to be necessary and sufficient for the convergence of products of stochastic matrices from a finite set.

3.2.1. Theorem. *[6] Let $\mathcal{P}$ be a finite set of stochastic matrices. For each sequence of matrices $P(1),\ P(2),\ P(3), \ldots$ from $\mathcal{P}$, $P(k) \cdots P(1)$ converges to a rank-one matrix $\mathbf{1}c^T$ as $k \to \infty$ if and only if any of the three conditions C1, C2 or C3 holds.*

Instead of *finite* sets of stochastic matrices, in [26] *compact* sets of stochastic matrices are studied and conditions C1 to C3 are shown to be sufficient for the convergence of products of stochastic matrices.

3.2.2. PROPOSITION. *Let $\mathcal{P}$ be a compact set of stochastic matrices. For each sequence of matrices $P(1),\ P(2),\ P(3),\ldots$ from $\mathcal{P}$, $P(k)\cdots P(1)$ converges to a rank-one matrix $\mathbf{1}c^T$ as $k \to \infty$ if any of the three conditions C1, C2 or C3 holds.*

In what follows, we first show that conditions C1 to C3 are also necessary for the convergence and then further construct an additional equivalent condition using the notion of the Sarymsakov class defined in Definition 2.3.4.

3.2.3. PROPOSITION. *Let $\mathcal{P}$ be a compact set of stochastic matrices. For each sequence of matrices $P(1),\ P(2),\ P(3),\ldots$ from $\mathcal{P}$, $P(k)\cdots P(1)$ converges to a rank-one matrix $\mathbf{1}c^T$ as $k \to \infty$ only if any of the three conditions C1, C2 or C3 holds.*

Proof. Since C1, C2 and C3 are equivalent, it suffices to show that C1 is necessary. For each integer $l \geqslant 1$, we define $B = P(l)\cdots P(1)$ and consider the following converging matrix product $\cdots B \cdots BB$, which implies the fact that B^m converges to a rank-one matrix as the integer m goes to infinity. In view of the definition for SIA matrices, we know that B must be SIA for each $l \geqslant 1$. □

Now we consider the following condition.

C4. There is an integer $\alpha \geqslant 1$ such that for each $k \geqslant \alpha$ and any $P(i) \in \mathcal{P}$, $1 \leqslant i \leqslant k$, the matrix $P(k)\cdots P(1)$ belongs to the Sarymsakov class $\mathcal{K}$.

One can further prove the following relationship between C4 and the other three conditions.

3.2.4. PROPOSITION. *C4 is equivalent to C1, C2 and C3.*

Proof. It suffices to show that C4 is equivalent to C2. Suppose C4 holds. Since the product of $N-1$ matrices from the Sarymsakov class is scrambling [89], if we take $\nu = (N-1)\alpha$, then $P(k)\cdots P(1)$ is scrambling for $k \geqslant \nu$. So C4$\Longrightarrow$C2. On the other hand, since a scrambling matrix always belongs to the Sarymsakov class, we have C2$\Longrightarrow$C4. Hence, combining the two facts, we know C2$\Longleftrightarrow$C4. □

In view of Propositions 3.2.2, 3.2.3 and 3.2.4, we have in fact proved the following theorem.

3.2.2. THEOREM. *Let $\mathcal{P}$ be a compact set of stochastic matrices. For each sequence of matrices $P(1),\ P(2),\ P(3),\ldots$ from $\mathcal{P}$, $P(k)\cdots P(1)$ converges to a rank-one matrix $\mathbf{1}c^T$ as $k \to \infty$ if and only if any of the four conditions C1, C2, C3 or C4 holds.*

We want to further comment that, using the technique in [101], one can prove the same conclusion in Theorem 3.2.2 holds when the positive elements of the matrices in $\mathcal{P}$ have a positive lower bound. In other words, instead of requiring the set $\mathcal{P}$ to be compact, one may require that there is a uniform lower bound $\gamma > 0$ for all nonzero elements $p_{ij}(k)$ of $P(k)$, i.e., $p_{ij}(k) \geqslant \gamma > 0$ for all $k \geqslant 1$.

The condition C3 is closely related to the notion of "sequential connectivity", which is defined in [5] to study the convergence rate of consensus algorithms. Given a graph $\mathbb{G} = (\mathcal{V}, \mathcal{E})$ with N vertices, assume that self-loops are allowed here. For a set $\mathcal{A} \subseteq \mathcal{N} = \{1, \ldots, N\}$, we use $\mathcal{N}(\mathcal{A}, \mathbb{G}) = \{j : \exists\ i \in \mathcal{A},\ (v_i, v_j) \in \mathcal{E}\}$ to denote the set of indices of the neighbors of vertices in $\mathcal{A}$. We say a sequence of graphs $\mathbb{G}_i,\ i = 1, \ldots, T$, with a common vertex set $\mathcal{V} = \{v_1, \ldots, v_N\}$ is *sequentially connected* if there exists a vertex $v_j \in \mathcal{V}$ and a sequence of sets $\mathcal{A}_i \subseteq \mathcal{N}$ such that $\mathcal{A}_1 = \{j\},\ \mathcal{A}_{k+1} \subseteq \mathcal{N}(\mathcal{A}_k, \mathbb{G}_k)$ for all $1 \leqslant k \leqslant T - 1$ and $\mathcal{A}_T = \mathcal{N}$.

We associate a directed, weighted graph $\mathbb{G}(P)$ to the stochastic matrix P, which is defined in Definition 2.2.1 except that self-loops are allowed in the graph $\mathbb{G}$; that is there is an edge $(v_i, v_i) \in \mathcal{E}$ if and only if $p_{ii} \neq 0$. Denote the graphs associated with the stochastic matrix $P(i)$ by $\mathbb{G}_i$. From the definition, one can easily see that the matrix product $P(k) \cdots P(1)$ has a positive column if and only if the corresponding sequence of graphs $\mathbb{G}_1,\ \mathbb{G}_2, \ldots, \mathbb{G}_k$ is sequentially connected.

In this section, we have reviewed the most important results on the convergence of products of stochastic matrices. In addition, we have focused on the Sarymsakov class to construct a new necessary and sufficient condition for the convergence. In the next section, we will look at how to apply these matrix theoretic ideas to the study of asynchronous system (3.5).

3.3 Coordinating multi-agent systems with asynchronous updates

We first provide an example to show that when implementing a converging synchronous algorithm in (3.2) asynchronously, the resulting asynchronous algorithm may not converge any more.

3.3.1. Example. Consider a converging synchronous algorithm in the form of (3.2) that is characterized by the matrix

$$P = \begin{bmatrix} 0 & 0 & 0 & 0.5 & 0.5 \\ 1 & 0 & 0 & 0 & 0 \\ 0.5 & 0.5 & 0 & 0 & 0 \\ 0 & 0.5 & 0.5 & 0 & 0 \\ 0 & 0 & 0.5 & 0.5 & 0 \end{bmatrix}. \tag{3.8}$$

One can check that P belongs to the Sarymsakov class $\mathcal{K}$ defined in Definition 2.3.4. Now we check the case when the agents update periodically with the update sequence of $\{2, 3, 1, 4, 5\}$ in each update period. Then one has

$$P_5 P_4 P_1 P_3 P_2 = \begin{bmatrix} 0 & 0 & 0 & 0.5 & 0.5 \\ 1 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 \end{bmatrix},$$

for which -1 is an eigenvalue, and this implies that $P_5 P_4 P_1 P_3 P_2 \notin \mathcal{K}$ and it is not even SIA anymore. So in this example, if all the agents update periodically, even given $P \in \mathcal{K}$, the system (3.2) after asynchronous implementation does not converge any more. □

The above example highlights the difference between synchronous and asynchronous systems. Thus it will be of great interest if one can identify those classes of synchronous algorithms that can still converge and achieve the algorithms' design objectives even after they are implemented asynchronously. Towards this end, we now turn our attention to a subclass of the Sarymsakov class $\mathcal{K}$, the set of scrambling matrices, and show that if the stochastic matrix P is scrambling, then the states of all the agents still become the same asymptotically if the algorithm described by (3.2) is implemented asynchronously. The following theorem summarizes our main result in this section.

3.3.1. Theorem. *If the stochastic matrix P in (3.2) is scrambling and there exists an infinite sequence of contiguous, nonempty, uniformly bounded time-intervals $[t_i, t_{i+1})$, starting at $t_0 = 0$ with $t_i \in \mathcal{T},\ i \geqslant 0$ and the property that across each such interval every agent updates at least once, then with asynchronous implementation the states of all the agents in system (3.5) become the same asymptotically.*

The proof of the theorem makes use of the following lemmas. For an arbitrary vector $x = [x_1, x_2, \ldots, x_N]^T$, define $\bar{x} = \max_{1 \leqslant i \leqslant N} x_i$ and $\underline{x} = \min_{1 \leqslant i \leqslant N} x_i$.

3.3.1. Lemma. *Let $P = \{p_{ij}\}_{N \times N}$ be an arbitrary stochastic matrix.*
(a) [90] *Let $x = [x_1, x_2, \ldots, x_N]^T$ be an arbitrary vector. If $z = Px$, then*

$$\bar{z} - \underline{z} \leqslant \tau(P)(\bar{x} - \underline{x}), \tag{3.9}$$

where $\tau(P) = \frac{1}{2} \max_{l,k} \sum_{s=1}^{N} |p_{ls} - p_{ks}|$ is the coefficient of ergodicity of P defined in Definition 2.3.6.
(b) There exists a vector x^* satisfying $\bar{x}^* > \underline{x}^*$ such that if $z^* = Px^*$, then

$$\bar{z}^* - \underline{z}^* = \tau(P)(\bar{x}^* - \underline{x}^*). \tag{3.10}$$

Proof: (a) is Theorem 3.1 in [90].
(b) Without loss of generality, assume $\tau(P) = \frac{1}{2}\sum_{s=1}^{N}|p_{is} - p_{js}|$. Let $u_l = p_{il} - p_{jl}$, $1 \leqslant l \leqslant N$, let L' denote the set of indices for which $u_l > 0$ if $l \in L'$, and let L'' denote the set of indices for which $u_l < 0$ if $l \in L''$. Note that $\sum_{l=1}^{N} u_l = 0$ and $\tau(P) = \frac{1}{2}(\sum_{l\in L'} u_l - \sum_{l\in L''} u_l) = \sum_{l\in L'} u_l$. Since L' is empty only if $u = [u_1, \ldots, u_n]^T = 0$, one has $\tau(P) = 0$ in this case. (3.10) holds for any vector x^* satisfying $\bar{x}^* > \underline{x}^*$.

Assume L' is not empty. Pick x^* such that

$$x_l^* = \begin{cases} 1, & l \in L' \\ -1, & l \in L'' \\ 0, & \text{otherwise.} \end{cases}$$

One has

$$z_i^* - z_j^* = \sum_{l=1}^{N} u_l x_l^* = \sum_{l\in L'} u_l - \sum_{l\in L''} u_l = 2\sum_{l\in L'} u_l = 2\tau(P) = \tau(P)(\bar{x}^* - \underline{x}^*).$$

Combining with (3.9), (3.10) holds. □

3.3.2. Lemma. *Let $P = \{p_{ij}\}_{N\times N}$ be an arbitrary stochastic matrix. If for any vector $x \in \mathbb{R}^N$ satisfying $\bar{x} > \underline{x}$ and $z = Px$, $\bar{z} - \underline{z} < \bar{x} - \underline{x}$, then P is scrambling.*

Proof: From Lemma 3.3.1(b), there exists a vector x^* satisfying $\bar{x}^* > \underline{x}^*$ such that $z^* = Px^*$ and $\bar{z}^* - \underline{z}^* = \tau(P)(\bar{x}^* - \underline{x}^*)$. Thus one has $\tau(P)(\bar{x}^* - \underline{x}^*) < \bar{x}^* - \underline{x}^*$, which implies $\tau(P) < 1$. One can conclude that P is scrambling from Lemma 2.3.1. □

3.3.3. Lemma. *Let $P = \{p_{ij}\}_{N\times N}$ be a stochastic scrambling matrix and let $c_1, c_2, \ldots, c_s$ be a finite sequence of indices from $\{1, 2, \ldots, N\}$ satisfying that for any $j \in \{1, 2, \ldots, N\}$, there exists an index l, $1 \leqslant l \leqslant s$ such that $c_l = j$.*
(a) Let $x \in \mathbb{R}^N$ be an arbitrary vector satisfying $\bar{x} > \underline{x}$ and $z = P_{c_s}\cdots P_{c_2}P_{c_1}x$ with P_{c_l} defined in (3.6) for $1 \leqslant l \leqslant s$. $\bar{z} - \underline{z} < \bar{x} - \underline{x}$.
(b) $P_{c_s}\cdots P_{c_2}P_{c_1}$ is scrambling.

Proof: (a) For an arbitrary vector $x = [x_1, \ldots, x_N]^T$, assume that $x_{a_1} = \cdots = x_{a_r} = \underline{x}$, $x_{b_1} = \cdots = x_{b_m} = \bar{x}$, where $a_1, \ldots, a_r, b_1, \ldots, b_m \in \{1, \ldots, N\}$ and define $\mathcal{A} = \{a_1, a_2 \ldots, a_r\}$, $\mathcal{B} = \{b_1, b_2, \ldots, b_m\}$. Define

$$\begin{aligned} &x(1) = x, \\ &x(t+1) = P_{c_t}x(t) = P_{c_t}[x_1(t), \ldots, x_N(t)]^T,\ 1 \leqslant t \leqslant s. \end{aligned}$$

Thus $z = x(s+1) = P_{c_s} \cdots P_{c_2} P_{c_1} x$. Since the set $\{1, 2, \ldots, N\}$ has N elements, we associate $x_1(t), x_2(t), \ldots, x_N(t)$ with N agents and accordingly call them the states of agents $1, 2, \ldots, N$ at step t. In view of the fact that $x(t+1) = P_{c_t} x(t)$ and the structure of the stochastic matrix P_{c_t} defined in (3.6), we know that only one element of $x(t+1)$ could be possibly different from that of $x(t)$ and

$$\bar{x}(t+1) \leqslant \bar{x}(t) \leqslant \bar{x}, \quad \underline{x}(t+1) \geqslant \underline{x}(t) \geqslant \underline{x}. \tag{3.11}$$

Assume that there exists an index w, $1 \leqslant w \leqslant s+1$ such that $c_w = i$, $x_i(w+1)$ is equal to $\bar{x}$ or $\underline{x}$, and for all $1 \leqslant t < w$ and $c_t = k$, $\underline{x} < x_k(t+1) < \bar{x}$.

Without loss of generality, assume $c_w = i \notin \mathcal{A} \cup \mathcal{B}$ and $x_i(w+1) = \underline{x}$. Since $\underline{x} < x_j(t) \leqslant \bar{x}$ for $j \notin \mathcal{A}$, $1 \leqslant t \leqslant w$, one has $p_{iu} = 0$, $u \notin \mathcal{A}$, and there exist indices $u_1, \ldots, u_l \in \mathcal{A}$ such that $p_{iu_k} > 0$ for $1 \leqslant k \leqslant l$. From Definition 2.3.5 of scrambling matrices, we know that for any $j \neq i$, there exists an index $v_j \in \{u_1, \ldots, u_l\}$ such that $p_{jv_j} > 0$. Thus for any $j \in \{1, \ldots, N\}$, there always exists an index $v_j \in \{u_1, \ldots, u_l\} \in \mathcal{A}$ such that $p_{jv_j} > 0$.

We now prove $x_j(t) < \bar{x}$ for $j \notin \mathcal{B}$ and $1 \leqslant t \leqslant s+1$. Since this is true for $1 \leqslant t \leqslant w$, one has

$$\begin{aligned} x_i(w+1) &= p_{iv_i} x_{v_i}(w) + \sum_{k \neq v_i} p_{ik} x_k(w) < p_{iv_i} \bar{x} + \sum_{k \neq v_i} p_{ik} \bar{x} = \bar{x}, \\ x_j(w+1) &= x_j(w) < \bar{x}, \quad j \neq i, \ j \notin \mathcal{B}. \end{aligned}$$

By induction, $x_j(t) < \bar{x}$ holds for $j \notin \mathcal{B}$ and $1 \leqslant t \leqslant s+1$.

From the assumption of the lemma, one knows that for any $j \in \mathcal{B}$, there always exists an index l, $1 \leqslant l \leqslant q$ such that $c_{t_l} = j$. In addition

$$x_j(t_l+1) = p_{jv_j} x_{v_j}(t_l) + \sum_{k \neq v_j} p_{jk} x_k(t_l) < \bar{x}.$$

Thus $x_j(t) < \bar{x}$ for $t \geqslant t_l + 1$, implying $x_j(s+1) < \bar{x}$, $j \in \mathcal{B}$. So $\bar{x}(s+1) < \bar{x}$.

Similarly, we can also arrive at the conclusion that $\bar{x}(s+1) < \bar{x}$ when $i \in \mathcal{A} \cup \mathcal{B}$, which implies $\bar{z} - \underline{z} < \bar{x} - \underline{x}$. If $x_i(w+1) = \bar{x}$, similar arguments show that $\underline{x}(s+1) < \underline{x}$, which also implies $\bar{z} - \underline{z} < \bar{x} - \underline{x}$.

If such an index w does not exist, then we have that for all $1 \leqslant t \leqslant s$ and $c_t = j$, $\underline{x} < x_j(t+1) < \bar{x}$. Since for any $j \in \{1, 2, \ldots, N\}$, there exists an index l, $1 \leqslant l \leqslant q$ satisfying $c_{t_l} = j$, one knows that $\underline{x} < x_j(s+1) < \bar{x}$ for all $1 \leqslant j \leqslant N$. Thus $\bar{z} - \underline{z} < \bar{x} - \underline{x}$, which completes the proof.

(b) It is a direct consequence of (a) and Lemma 3.3.2. □

Proof of Theorem 3.3.1: We denote the agent which updates at time t_i by c_{t_i}. The sequence c_{t_i}, $c_{t_i+1}, \ldots, c_{t_{i+1}-1}$ is a finite sequence of indices satisfying the condition in Lemma 3.3.3. One has $P_{c_{t_{i+1}-1}} \cdots P_{c_{t_i+1}} P_{c_{t_i}}$ is scrambling for all $i \geqslant 0$. In view of the uniformly boundedness of the intervals $[t_i, t_{i+1})$ and the fact that if one or more matrices in a product of matrices is scrambling, so is the product [40], we know that the product of the system's matrices with asynchronous implementation can be written as products of scrambling matrices from a finite set. Thus by applying Theorem 3.2.2, we arrive at the conclusion. □

The preceding discussions assume that no two or more agents update at the same time. In fact if there are two or more agents updating exactly at the same time, the conclusion in Theorem 3.3.1 still holds. To see this, we consider at time t, agents $c_1, c_2, \ldots, c_k$ update, where $1 \leqslant c_i \leqslant N$. Similar to Eqs. (3.3) and (3.4), the states of these agents satisfy

$$x_{c_l}(t+1) = \sum_{j=1}^{N} p_{c_l j} x_j(t),\ l = 1, \ldots, k, \tag{3.12}$$

and correspondingly for all the other agents, we have

$$x_j(t+1) = x_j(t),\ j \in \{1, \ldots, N\} \backslash \{c_1, \ldots, c_k\}. \tag{3.13}$$

Rewrite (3.12) and (3.13) in a compact form

$$x(t+1) = P_C x(t),$$

where

$$P_C = \begin{bmatrix} 1 & \cdots & 0 & 0 & \cdots & 0 \\ \vdots & \ddots & \vdots & \vdots & \cdots & \vdots \\ 0 & \cdots & 1 & 0 & \cdots & 0 \\ p_{c_1 1} & \cdots & p_{c_1, c_1-1} & p_{c_1 c_1} & \cdots & p_{c_1 N} \\ \vdots & \cdots & \cdots & \vdots & \ddots & \vdots \\ p_{c_k 1} & \cdots & p_{c_k, c_1-1} & p_{c_k c_1} & \cdots & p_{c_k N} \\ 0 & \cdots & 0 & 0 & 0 & 1 \end{bmatrix}. \tag{3.14}$$

For a set $C = \{c_1, \ldots, c_k\} \subseteq \{1, 2, \ldots, N\}$, we define the matrix P_C as in (3.14). We have a similar result to Lemma 3.3.3, based on which the correctness of Theorem 3.3.1 can be shown when two or more agents could possibly update at the same time.

3.3.4. Lemma. *If $P = \{p_{ij}\}_{N \times N}$ is a stochastic scrambling matrix and $C_1 = \{c_{11}, \ldots, c_{1k_1}\}, C_2 = \{c_{21}, \ldots, c_{2k_2}\}, \ldots, C_s = \{c_{s1}, \ldots, c_{sk_s}\}$ is a finite sequence of subsets of $\{1, 2, \ldots, N\}$ satisfying that for any $j \in \{1, 2, \ldots, N\}$, there exists an index l, $1 \leqslant l \leqslant s$ such that $j \in C_l$, then $P_{C_s} \cdots P_{C_2} P_{C_1}$ is scrambling with P_{C_i} defined in (3.14).*

The idea of the proof of this lemma is exactly the same as that of Lemma 3.3.3. Hence we omit it here.

Although the synchronous coordination algorithm specified in Theorem 3.3.1 still converges after asynchronous implementation, the convergence rate changes. In the next section, we look into the performances of asynchronous coordination algorithms through simulations.

3.4 Illustrative example

In this section, we perform simulation studies. We consider the case when the matrix P is a scrambling stochastic matrix. Let

$$P = \begin{bmatrix} \frac{1}{4} & \frac{1}{4} & 0 & \frac{1}{4} & 0 & \frac{1}{4} & 0 & 0 & 0 & 0 \\ \frac{1}{4} & 0 & \frac{1}{4} & 0 & 0 & 0 & \frac{1}{4} & 0 & 0 & \frac{1}{4} \\ 0 & \frac{1}{4} & 0 & \frac{1}{4} & 0 & \frac{1}{4} & \frac{1}{4} & 0 & 0 & 0 \\ 0 & \frac{1}{4} & \frac{1}{4} & \frac{1}{4} & \frac{1}{4} & 0 & 0 & 0 & 0 & 0 \\ 0 & \frac{1}{3} & 0 & 0 & 0 & \frac{1}{3} & \frac{1}{3} & 0 & 0 & 0 \\ 0 & 0 & 0 & \frac{1}{3} & \frac{1}{3} & 0 & \frac{1}{3} & 0 & 0 & 0 \\ \frac{1}{4} & \frac{1}{4} & 0 & \frac{1}{4} & 0 & 0 & 0 & 0 & \frac{1}{4} & 0 \\ 0 & 0 & \frac{1}{4} & 0 & \frac{1}{4} & \frac{1}{4} & 0 & 0 & \frac{1}{4} & 0 \\ 0 & 0 & 0 & 0 & \frac{1}{4} & \frac{1}{4} & 0 & 0 & \frac{1}{4} & \frac{1}{4} \\ 0 & 0 & 0 & \frac{1}{3} & 0 & 0 & \frac{1}{3} & 0 & \frac{1}{3} & 0 \end{bmatrix}.$$

According to Theorem 3.3.1, the algorithm after asynchronous implementation still converges since P is scrambling. We then carry out simulations to compare the performances of the algorithm that runs synchronously and asynchronously. Take the initial values of the ten agents to be $x_i(0) = i, \ i = 1, \ldots, 10$. When we implement the algorithm synchronously, the converging process is shown in Fig. 3.1. Now consider the case when the algorithm runs asynchronously. Assume at every time instant a node i is chosen randomly among the ten agents with probability $\frac{1}{10}$. Then agent i updates its state according to Eq. (3.3) and the rest agents keep the states unchanged. The evolution of the states of all the agents is illustrated in Fig. 3.2, from which it is easy to see that the convergence process slows down compared to that in Fig. 3.1.

3.5 Conclusion

In this chapter, we have reviewed some classic and recent results on backward products of stochastic matrices and developed some new necessary and sufficient conditions for convergence using the Sarymsakov class. A sufficient condition has been

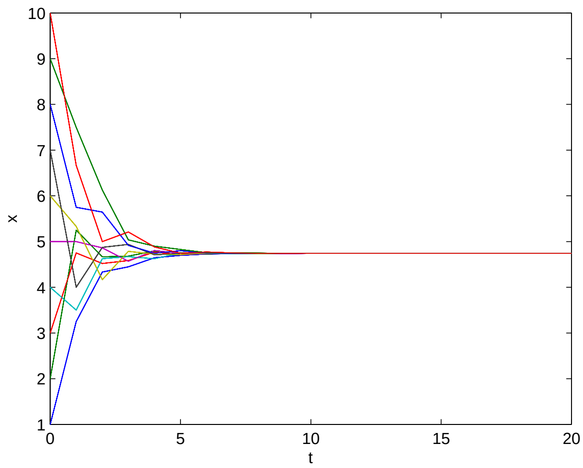


Figure 3.1: *The evolution of the agents' states when the agents update synchronously.*

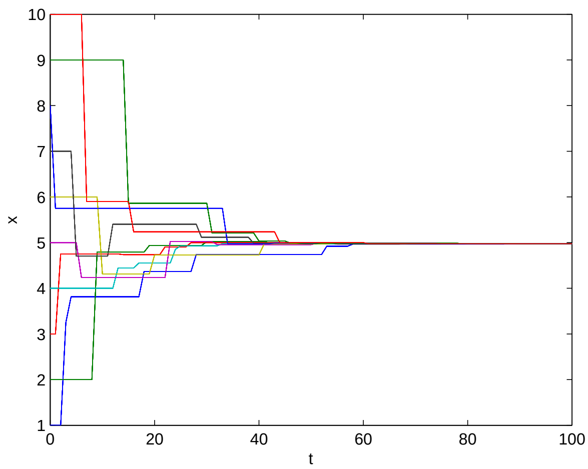


Figure 3.2: *The evolution of the agents' states when the agents update asynchronously.*

constructed to guarantee that a coordination algorithm, which converges when implemented synchronously, still converges when it is implemented asynchronously. The condition is stipulated using the notion of scrambling stochastic matrices, which all belong to the Sarymsakov class.

There are still open questions in order to understand better the asynchronous implementation of distributed coordination algorithms. It is even more challenging to study the case when each agent cannot update their states instantaneously, in which case the process of analytic synchronization [54, 15] is extremely difficult to carry out. Tools from the analysis of hybrid systems may turn out to be helpful.

Chapter 4

Distributed algorithms with positive and negative couplings

Much of the work on distributed algorithms has assumed that all the agents in a network are working cooperatively to reach an agreement. However, as pointed out in Chapter 1, several typical networks suggest that it is more reasonable to assume that the interaction between a pair of agents in a network can be either cooperative or competitive. Very recent results [2, 3] have shown that polarization of the states of the agents may arise in this case by employing the notion of structural balance from social network theory. In this chapter, we study distributed algorithms in the presence of positive and negative couplings with an emphasis on the case when the network topologies are time-varying. It is shown that the states of the agents polarize under some connectivity conditions if all the networks involved are structurally balanced and they maintain a common bipartition of two opposing factions. If structurally unbalanced networks arise often enough as time evolves, then the states of all the agents asymptotically agree and converge to zero.

4.1 Problem formulation

In Section 2.4, we have reviewed several well-studied distributed algorithms, in which the weights $p_{ij}(t)$ in system (2.8) or $a_{ij}(t)$ in system (2.11) are assumed to be nonnegative. In this chapter, we generalize the models (2.8) and (2.11) by taking negative couplings between the agents into account. Consider a multi-agent system consisting of N agents and each agent $i,\ i = 1, \ldots, N$, has a real value x_i. In the discrete-time setting, the values of the agents are updated according to

$$x_i(t+1) = \sum_{j=1}^{N} p_{ij}(t)x_j(t), \qquad t = 0, 1, \ldots. \tag{4.1}$$

or in a compact form

$$x(t+1) = P(t)x(t), \qquad t = 0, 1, \ldots. \tag{4.2}$$

where $x_i(t) \in \mathbb{R}$, $x(t) = [x_1(t), \ldots, x_N(t)]^T$, $P(t) = (p_{ij}(t))_{N\times N} \in \mathbb{R}^{N\times N}$ and $p_{ij}(t)$ is the weight agent i assigned to agent j when agent i updates its state at time t. In contrast to system (2.8), in which $p_{ij}(t)$ are assumed to be nonnegative, here the weights $p_{ij}(t)$ can also be negative, which characterize the antagonistic relationship between individuals in a social network. Signed graphs are used to model networks with positive and negative couplings among the agents. The weights $p_{ij}(t)$ in system (4.1) satisfy the following condition

$$\sum_{j=1}^{N} |p_{ij}(t)| = 1, \quad p_{ii}(t) > 0, \ i = 1, \ldots, N. \tag{4.3}$$

This is an extension of the assumption (2.10) in Chapter 2 on the weights in distributed averaging algorithm (2.8), where $p_{ij}(t)$ are assumed to be nonnegative and $\sum_{j=1}^{N} p_{ij}(t) = 1,\ i = 1, \ldots, N$.

The continuous-time counterpart of the distributed algorithm (4.1) is given by

$$\dot{x}_i = -\sum_{j=1}^{N} |a_{ij}(t)|(x_i - sgn(a_{ij}(t))x_j), \qquad i = 1, \ldots, N, \tag{4.4}$$

where $a_{ij}(t) \in \mathbb{R}$ is the ijth entry of $A(t)$, $A(t)$ is the signed adjacency matrix of the signed graph $\mathbb{G}(t)$ representing the interaction topology at time t, and $sgn(\cdot)$ is the sign function. Let the signed Laplacian matrix $L(t) = (l_{ij}(t))_{N\times N}$ be given by (2.3). System (4.4) can be written in a compact form

$$\dot{x} = -L(t)x. \tag{4.5}$$

In contrast to system (2.11), where $a_{ij}(t)$ are assumed to be nonnegative, here the weights $a_{ij}(t)$ can also be negative.

4.1.1. Definition. *System (4.2) or system (4.5) admits a polarization if for all initial value,* $\lim_{t\to\infty} |x_i(t)| = \alpha,\ i = 1, \ldots, N$, *and there exists an initial value* $x(0)$ *such that* $\lim_{t\to\infty} |x_i(t)| = \alpha > 0,\ i = 1, \ldots, N$.

When system (4.2) or system (4.5) admits a polarization, it often happens that the agents in the network split into two clusters and the agents in the same cluster hold the same asymptotic value while the agents in different clusters hold opposite values. We have seen that for distributed averaging algorithms in [47, 81, 16], the states of the agents asymptotically converge to a common value under some connectivity conditions. Definition 4.1.1 includes this phenomenon as a special case.

To study the dynamical behavior of system (4.5) under fixed interaction topology, the notion of structural balance in social network theory has been employed in [3], which will be introduced in the next section. It is shown that in a structurally balanced network that can be partitioned into two opposing factions, the states of the agents in the same faction converge to the same value, while the states of the agents in different factions converge to two opposite values asymptotically; in a structurally unbalanced network, the states of all the agents asymptotically agree and converge to zero. What is more intriguing is to investigate the dynamical behaviors of system (4.2) and system (4.5) under time-varying interaction topologies, since in practical situations the relationship between agents may change with time. This also brings great challenge to the theoretical analysis. We are interested in finding out whether polarization will arise or agreement can be reached in system (4.2) and system (4.5) under dynamically changing interaction topologies.

4.2 Distributed discrete-time algorithms

4.2.1 Discrete-time updates under fixed topologies

In this section, we consider the case when the interaction topology is time-invariant, that is $p_{ij}(t)$ is time-invariant and simply denoted by p_{ij}. Systems (4.1) and (4.2) become

$$x_i(t+1) = \sum_{j=1}^{N} p_{ij} x_j(t), \qquad t = 0, 1, \ldots, \tag{4.6}$$

and

$$x(t+1) = Px(t), \qquad t = 0, 1, \ldots. \tag{4.7}$$

Assumption (4.3) becomes

$$\sum_{j=1}^{N} |p_{ij}| = 1, \quad p_{ii} > 0, \ i = 1, \ldots, N. \tag{4.8}$$

Let $\mathbb{G}(P)$ be the signed graph associated with P representing the interaction topology. A cycle in $\mathbb{G}$ is said to be *positive* if it contains an even number of negative weights; a *negative cycle* is not positive. We first introduce the notion of structural balance [41, 22] in signed graphs.

4.2.1. Definition. *A signed graph $\mathbb{G}(P) = (\mathcal{V}, \mathcal{E})$ is structurally balanced if there is a bipartition $\{\mathcal{V}_1, \mathcal{V}_2\}$ of $\mathcal{V}$, $\mathcal{V}_1 \cup \mathcal{V}_2 = \mathcal{V}$, $\mathcal{V}_1 \cap \mathcal{V}_2 = \varnothing$ such that $p_{ij} \geqslant 0$, $\forall v_i, v_j \in \mathcal{V}_k$, $k \in \{1,2\}$ and $p_{ij} \leqslant 0$, $\forall v_i \in \mathcal{V}_k, v_j \in \mathcal{V}_l$, $k \neq l$, $k, l \in \{1,2\}$; it is structurally unbalanced otherwise.*

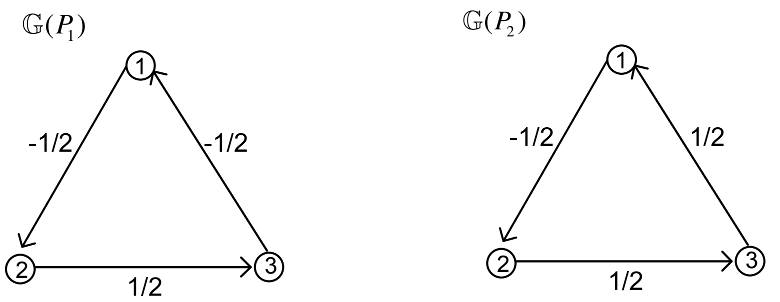


Figure 4.1: *Signed graphs $\mathbb{G}(P_1)$ and $\mathbb{G}(P_2)$ associated with P_1 and P_2. $\mathbb{G}(P_1)$ is structurally balanced and $\mathbb{G}(P_2)$ is structurally unbalanced.*

4.2.1. Example. Let

$$P_1 = \begin{bmatrix} 1/2 & 0 & -1/2 \\ -1/2 & 1/2 & 0 \\ 0 & 1/2 & 1/2 \end{bmatrix}, \; P_2 = \begin{bmatrix} 1/2 & 0 & -1/2 \\ 1/2 & 1/2 & 0 \\ 0 & 1/2 & 1/2 \end{bmatrix}.$$

Signed graphs $\mathbb{G}(P_1)$ and $\mathbb{G}(P_2)$ associated with P_1 and P_2 are illustrated in Fig. 4.1. It is obvious from Definition 4.2.1 that $\mathbb{G}(P_1)$ is structurally balanced, while $\mathbb{G}(P_2)$ is structurally unbalanced. The only cycle in $\mathbb{G}(P_1)$ is positive and the one in $\mathbb{G}(P_2)$ is negative. The eigenvalues of P_1 are 1, $0.25 \pm 0.4330\imath$, and those of P_2 are 0, $0.75 \pm 0.4330\imath$, where $\imath$ is the imaginary unit. Note that 1 is an eigenvalue of P_1, while all the eigenvalues of P_2 are located inside the unit disc. □

We also say that the matrix P is structurally balanced if its associated graph $\mathbb{G}(P)$ is structurally balanced. For convenience, in this chapter, we use $||x||$ to denote the max norm $||x||_\infty$ of a vector $x \in \mathbb{R}^N$. For an arbitrary matrix $A = (a_{ij})_{N\times N}$, let $|A| = (|a_{ij}|)_{N\times N}$. For a matrix P satisfying (4.8), $|P|$ is a stochastic matrix and 1 is always an eigenvalue of $|P|$. The Gersgorin discs $G(P)$ are all contained in the unit disc, 1 is on the boundary of $G(P)$ and -1 is not inside the Gersgorin region. Thus in view of Lemma 2.2.1, -1 is not an eigenvalue of P. The following result can be further used to determine whether a boundary point is an eigenvalue.

4.2.1. Lemma. *[44] Let $A = (a_{ij})_{N\times N}$ and suppose that λ is an eigenvalue of A that is a boundary point of $G(A)$, or, more generally, satisfies the inequalities*

$$|\lambda - a_{ii}| \geqslant \sum_{j=1, j\neq i}^{N} |a_{ij}|, \quad i = 1, \ldots, N, \tag{4.9}$$

If A is irreducible, then
(a) every Gersgorin circle passes through λ; and
(b) if $Ax = \lambda x$ and $x = [x_1, \dots, x_N]^T \neq 0$, then $|x_i| = |x_j|$ for all $i, j = 1, \dots, N$.

For two arbitrary matrices $A,\ B \in \mathbb{R}^{N\times N}$, the following holds

$$|\sum_{k=1}^{N} a_{ik}b_{kj}| \leqslant \sum_{k=1}^{N} |a_{ik}||b_{kj}|. \tag{4.10}$$

This implies that $|AB| \leqslant |A||B|$. This result naturally extends to the case when more than two matrices are involved.

4.2.2. Lemma. *Let $A_i \in \mathbb{R}^{N\times N},\ i = 1, \dots, n$. Then $|A_n \cdots A_1| \leqslant |A_n| \cdots |A_2||A_1|$.*

The following two lemmas establish the relationship between the spectral radii of two matrices.

4.2.3. Lemma. *[44] Let $A,\ B \in \mathbb{R}^{N\times N}$. If $|A| \leqslant B$, then $\rho(A) \leqslant \rho(|A|) \leqslant \rho(B)$.*

4.2.4. Lemma. *[44] Let A be an irreducible nonnegative matrix. If $B \geqslant 0$ and $B \neq 0$ then $\rho(A + B) > \rho(A)$.*

Now we are ready to study the asymptotic behavior of system (4.7).

4.2.5. Lemma. *Let P be an irreducible matrix satisfying the condition (4.8). Its associated graph $\mathbb{G}(P)$ is structurally balanced if and only if one of the following equivalent conditions holds:*
(a) All cycles of $\mathbb{G}(P)$ are positive;
(b) There exists a diagonal matrix U satisfying $U^2 = I$ such that UPU is nonnegative; U is unique in the sense that if there exist two diagonal matrices $U_1,\ U_2$, satisfying $U_1^2 = I,\ U_2^2 = I$ such that $U_1PU_1, U_2PU_2 \geqslant 0$, then $U_1 = U_2$ or $U_1 = -U_2$;
(c) 1 is an eigenvalue of P.

Proof. (i-ii). Note that if $\mathbb{G}(P)$ is structurally balanced, then $p_{ij}p_{ji} \geqslant 0,\ i \neq j,\ i, j = 1, \dots, N$. Mimicking the proof of Lemma 2 in [3], we can show that (a) and (b) are equivalent conditions to structural balance.
(iii). (b)$\Longrightarrow$(c): Since UPU and P have the same eigenvalues and UPU is a stochastic matrix, 1 is an eigenvalue of P.
(c)$\Longrightarrow$ $\mathbb{G}(P)$ is structurally balanced: Since P is irreducible, it follows from Lemma 4.2.1 that there exists an eigenvector x of 1 satisfying $|x_i| = |x_j| > 0,\ i, j = 1, \dots, N$.

Let $\mathcal{V}_1 = \{v_i |\ x_i > 0,\ i = 1, \ldots, N\}$ and $\mathcal{V}_2 = \{v_i |\ x_i < 0,\ i = 1, \ldots, N\}$. Then $\mathcal{V}_1 \cup \mathcal{V}_2 = \mathcal{V},\ \mathcal{V}_1 \cap \mathcal{V}_2 = \varnothing$, since $|x_i| > 0\ i = 1, \ldots, N$. From $x = Px$, one has

$$|x_i| = \left|\sum_{j=1}^{N} p_{ij} x_j\right| \leqslant \sum_{j=1}^{N} |p_{ij} x_j| = \sum_{j=1}^{N} |p_{ij}||x_j| = |x_i|. \tag{4.11}$$

This implies that $x_i p_{ij} x_j \geqslant 0$ for all $j = 1, \ldots, N$. Assume that $(v_k, v_l) \in \mathcal{E}$ and $v_k \in \mathcal{V}_1,\ v_l \in \mathcal{V}_2$. Taking $i = k$ in (4.11), we have that $x_k p_{kj} x_j \geqslant 0$ for all $j = 1, \ldots, N$. Since $x_k > 0$ and $x_l < 0$, hence $p_{kl} < 0$. Similarly, one can show that if $(v_k, v_l) \in \mathcal{E}$ and $v_k, v_l \in \mathcal{V}_s,\ s = 1, 2$, then $p_{ij} > 0$. We conclude that $\mathbb{G}(P)$ is structurally balanced. □

From Lemma 4.2.5, we immediately have the following corollary.

4.2.1. Corollary. *Let P be an irreducible matrix satisfying the condition (4.8). Its associated graph $\mathbb{G}(P)$ is structurally unbalanced if and only if one of the following equivalent conditions holds:*
(a) $\mathbb{G}(P)$ has at least one negative cycle;
(b) There does not exist a diagonal matrix U satisfying $U^2 = I$ such that UPU is nonnegative;
(c) $|\lambda(P)| < 1$.

The following theorem is a consequence of Lemma 4.2.5, Corollary 4.2.1 and Theorem 2.4.1.

4.2.1. Theorem. *Let P be an irreducible matrix satisfying the condition (4.8). System (4.7) admits a polarization if and only if the graph $\mathbb{G}(P)$ is structurally balanced. Furthermore, if U is a diagonal matrix satisfying $U^2 = I$ such that UPU is nonnegative, then the state of system (4.7) asymptotically converges to $\lim_{t\to\infty} x(t) = v^T U x(0) U \mathbf{1}$, where v is a left normalized eigenvector of UPU corresponding to 1 such that $v^T \mathbf{1} = 1$. If $\mathbb{G}(P)$ is structurally unbalanced, then $\lim_{t\to\infty} x(t) = 0$ for every initial value.*

Proof. Define $y = Ux$. In view of Lemma 4.2.5, the transformation induces a system $y(t+1) = UPUy(t)$ with nonnegative UPU, when P is structurally balanced. The asymptotic state of the system can be obtained from Theorem 2.4.1. □

4.2.2 Discrete-time updates under time-varying topologies

In this section, we consider the case when the interaction graph topologies are dynamically changing. Let $\mathcal{P} = \{P_1, P_2, \ldots, P_n\}$ be a finite set of matrices, where $P_i,\ i = 1, \ldots, n$, satisfy the condition (4.8).

4.2.2. THEOREM. *Assume that $P(t) \in \mathcal{P}$ and there is a diagonal matrix U satisfying $U^2 = I$ such that $UP(t)U \geqslant 0$ for all $t = 0, 1, 2, \ldots$. Assume that there exists an infinite sequence of contiguous, nonempty, uniformly bounded time intervals $[t_i, t_{i+1})$, $i \geqslant 0$, starting at $t_0 = 0$ with the property that across each time interval $[t_i, t_{i+1})$, the union of the graphs $\mathbb{G}(P(t))$ are strongly connected. Then system (4.2) admits a polarization and $\lim_{t\to\infty} x(t) = \alpha U \mathbf{1}$, where α is some constant.*

Proof. A transformation $y = Ux$ transforms system (4.2) to $y(t+1) = UP(t)Uy(t)$ with nonnegative matrices $UP(t)U$. Theorem 2.4.2 verifies the correctness of the theorem. □

If P_i, $i = 1, \ldots, n$, in $\mathcal{P}$ are all strongly connected and structurally balanced and further the unique bipartitions of $\mathcal{V}$ satisfying Lemma 4.2.5 are identical for $\mathbb{G}(P_i)$, $i = 1, \ldots, n$, then the assumptions in Theorem 4.2.2 are satisfied and the states of the agents will converge to two opposite values. In system (2.8), the weight-s are nonnegative and trivially $P(t)$ is structurally balanced. Theorem 4.2.2 is a generalization of previous results in [47, 81, 69].

4.2.3. THEOREM. *Let $P(t) \in \mathcal{P}$, $t = 0, 1, 2, \ldots$. There exists an infinite sequence of contiguous, nonempty, uniformly bounded time intervals $[t_i, t_{i+1})$, $i \geqslant 0$, starting at $t_0 = 0$ with the property that across each time interval $[t_i, t_{i+1})$, the union of the graphs are strongly connected and there does not exist a diagonal matrix U satisfying $U^2 = I$ such that $UP(t)U \geqslant 0$, $t_i \leqslant t < t_{i+1}$. Then system (4.2) converges to zero asymptotically.*

If for each time interval $[t_i, t_{i+1})$, there always exists some $t \in [t_i, t_{i+1})$, such that $P(t)$ is strongly connected and structurally unbalanced, then the conditions in Theorem 4.2.3 are satisfied and thus the state of the system converges to zero. Said differently, if structural unbalance arises in the network often enough, then polarization of the states of the agents will not happen and instead the agents in the network reach an agreement finally.

Before proving Theorem 4.2.3, we first prove several lemmas that will be useful in the proof for Theorem 4.2.3.

4.2.6. LEMMA. *Assume that P, Q satisfy the condition (4.8) and $\mathbb{G}(P) \cup \mathbb{G}(Q)$ is strongly connected. If there does not exist a diagonal matrix U satisfying $U^2 = I$ such that $UPU, UQU \geqslant 0$, then $\rho(PQ) < 1$.*

Proof. Let $R = PQ$ and $W = |P||Q|$. Then W is stochastic and irreducible, since $\mathbb{G}(P) \cup \mathbb{G}(Q)$ is strongly connected. Thus 1 is a simple eigenvalue of W and all the other eigenvalues lie inside the unit disc [81]. Since

$$r_{ij} = \sum_{k=1}^{N} p_{ik}q_{kj} \leqslant |\sum_{k=1}^{N} p_{ik}q_{kj}| = |r_{ij}| \leqslant \sum_{k=1}^{N} |p_{ik}||q_{kj}| = w_{ij}, \tag{4.12}$$

it follows that $R \leqslant |R| \leqslant W$. In view of Lemma 4.2.3, $\rho(R) \leqslant \rho(|R|) \leqslant \rho(W) = 1$.

If $W - |R| \neq 0$, the matrix $|R| + \epsilon(W - |R|)$ is nonnegative and irreducible for some positive scalar $0 < \epsilon < 1$, since $W - |R|$ is nonnegative and $W = W + (W - |R|)$ is irreducible. It follows from Lemma 4.2.4 that

$$\rho(R) \leqslant \rho(|R|) \leqslant \rho(|R| + \epsilon(W - |R|)) < \rho(W) = 1.$$

If $W - |R| = 0$, then it is easy to see from the inequality (4.12) that $p_{ik}q_{kj} \geqslant 0$ for all $k = 1, \ldots, N$, or $p_{ik}q_{kj} \leqslant 0$ for all $k = 1, \ldots, N$. Since $p_{ii},\ q_{ii} > 0$ and $r_{ii} = \sum_{k=1}^{N} p_{ik}q_{ki}$, one has $r_{ii} > 0$ for all $i = 1, \ldots, N$. From (4.12),

$$\sum_{j=1}^{N} |r_{ij}| \leqslant \sum_{j=1}^{N} w_{ij} = 1.$$

The Gersgorin region $G(R)$ of R is contained in the unit disc and the boundary points of $G(R)$ are all inside the unit disc except 1. We have to show that 1 is not an eigenvalue of R. Observing that for any $p_{ij} < 0$, one has that $r_{ij} < 0$ since $p_{ij}q_{jj} < 0$. Similarly, for any $p_{ij} > 0$, $r_{ij} > 0$. For any $q_{ij} < 0$, one has that $r_{ij} < 0$, and for any $q_{ij} > 0$, $r_{ij} > 0$. Suppose on the contrary 1 is an eigenvalue of R. R is thus structurally balanced from Lemma 4.2.5 and there exists a diagonal matrix U satisfying $U^2 = I$ such that $URU \geqslant 0$. Furthermore, since a subgraph of a structurally balanced graph is also balanced, it follows that P and Q are both structurally balanced and $UPU, UQU \geqslant 0$. This contradicts the assumption of this lemma, completing the proof. □

4.2.7. Lemma. *Assume that $P_1, \ldots, P_n$ satisfy (4.8) and $\cup_{i=1}^{n} \mathbb{G}(P_i)$ is strongly connected. If there does not exist a diagonal matrix U satisfying $U^2 = I$ such that $UP_iU \geqslant 0,\ i = 1, \ldots, n$, then $\rho(P_n \cdots P_2P_1) < 1$.*

Proof. Denote the ijth element of P_k by $(P_k)_{ij}$, let $R = P_n \cdots P_2P_1$ and let $W = |P_n| \cdots |P_2||P_1|$. In view of the fact that

$$\begin{aligned} |r_{ij}| &= \left| \sum_{k_1,\ldots,k_{n-1}=1,\ldots,N} (P_n)_{ik_{n-1}} \cdots (P_2)_{k_2k_1}(P_1)_{k_1j} \right| \\ &\leqslant \sum_{k_1,\ldots,k_{n-1}=1,\ldots,N} |(P_n)_{ik_{n-1}}| \cdots |(P_2)_{k_2k_1}||(P_1)_{k_1j}| = w_{ij}, \end{aligned} \tag{4.13}$$

and mimicking the proof of Lemma 4.2.6, one can prove the lemma. □

The following result is an immediate consequence of Lemma 4.2.7.

4.2.2. COROLLARY. *Assume that $P_1, \dots, P_n$ satisfy (4.8) and $\cup_{i=1}^n \mathbb{G}(P_i)$ is strongly connected. 1 is an eigenvalue of $P_n \cdots P_2 P_1$ if and only if there exists a diagonal matrix U satisfying $U^2 = I$ such that $UP_iU \geqslant 0,\ i = 1, \dots, n$.*

4.2.8. LEMMA. *Let $P,\ Q \in \mathbb{R}^{N\times N}$ and let $W,\ S \in \mathbb{R}^{N\times N}$ be two positive stochastic matrices. Assume that $\rho(P) < 1,\ \rho(Q) < 1,\ |P| \leqslant W$, and $|Q| \leqslant S$. Assume that if $|P| = W$, then $p_{ii} > 0$ for all $i = 1, \dots, N$ and if $|Q| = S$, then $q_{ii} > 0$ for all $i = 1, \dots, N$. There exists a constant $0 \leqslant \tau < 1$ such that for any $x \in \mathbb{R}^N$, if $z = QPx$, then $||z|| \leqslant \tau||x||$.*

Proof. Let x be a nonzero vector and $y = Px$. One has that $z = Qy$ and

$$y_i = \sum_{j=1}^N p_{ij}x_j \leqslant \left|\sum_{j=1}^N p_{ij}x_j\right| \leqslant \sum_{j=1}^N |p_{ij}x_j| \tag{4.14}$$

$$\leqslant \sum_{j=1}^N w_{ij}|x_j| \leqslant \sum_{j=1}^N w_{ij}||x|| = ||x||. \tag{4.15}$$

Thus $||y|| \leqslant ||x||$ and it follows that $||z|| \leqslant ||y|| \leqslant ||x||$.

It follows from (4.14) and (4.15) that $|y_i| = ||y|| = ||x||$ for some $i = 1, \dots, N$ if and only if $|x_j| = ||x||,\ |p_{ij}| = w_{ij},\ j = 1, \dots, N$, and $p_{ij}x_j \geqslant 0$ for all $j = 1, \dots, N$ or $p_{ij}x_j \leqslant 0$ for all $j = 1, \dots, N$. Thus if for some k, $|x_k| \neq ||x||$, then $||y|| < ||x||$ and $||z|| \leqslant ||y|| < ||x||$.

Next we consider the vector x with $|x_i| = ||x||$ for all $i = 1, \dots, N$. If $|P| \neq W$, then there exists some p_{kj} such that $|p_{kj}| < w_{kj}$. The first inequality in (4.15) is strict for $i = k$, and thus $|y_k| < ||x||$. If $|y_k| = ||y||$, then $||z|| \leqslant ||y|| < ||x||$; if $|y_k| < ||y||$, then in view of the fact that $z = Qy$ and Q is in the same position as P in $y = Px$, one has that $||z|| < ||y|| \leqslant ||x||$.

If $|P| = W$, we prove that there exists some k such that $|y_k| < ||x||$, from which we can arrive at the conclusion that $||z|| < ||x||$. Suppose on the contrary that $|y_i| = ||y|| = ||x||$ for all $i = 1, \dots, N$. It follows that for a given $i,\ i = 1, \dots, N$, $p_{ij}x_j \geqslant 0$ for all $j = 1, \dots, N$ or $p_{ij}x_j \leqslant 0$ for all $j = 1, \dots, N$. Define two sets $\mathcal{N}_1 = \{i|\ x_i > 0,\ i = 1, \dots, N\}$ and $\mathcal{N}_2 = \{i|\ x_i < 0,\ i = 1, \dots, N\}$. Then $\{\mathcal{N}_1,\ \mathcal{N}_2\}$ is a bipartition of $\{1, \dots, N\}$. For any $i \in \mathcal{N}_1$, if $j \in \mathcal{N}_1$, then $p_{ij} > 0$, since $p_{ii} > 0,\ x_i > 0$; if $j \in \mathcal{N}_2$, then $p_{ij} < 0$. Similarly, if $i, j \in \mathcal{N}_2$, then $p_{ij} > 0$; if $i \in \mathcal{N}_2,\ j \in \mathcal{N}_1$, then $p_{ij} < 0$. P is thus structurally balanced. Since $|P| = W$ and $p_{ii} > 0,\ i = 1, \dots, N$,

P satisfies the condition (4.8). From Lemma 4.2.5, 1 is an eigenvalue of P, which contradicts the fact that $\rho(P) < 1$.

We have proved that for any nonzero x, $||QPx|| < ||x||$, that is $\frac{||QPx||}{||x||} < 1$. In addition, one has that

$$\sup_{x\neq 0} \frac{||QPx||}{||x||} = \sup_{x^T x=1} \frac{||QPx||}{||x||} = \max_{x^T x=1} \frac{||QPx||}{||x||}.$$

The last equality holds since the set $\{x|x^T x = 1\}$ is compact. Letting $\tau = \max\limits_{x^T x=1} \frac{||QPx||}{||x||}$, it is true that $\tau < 1$. We complete the proof. □

It is known that if $A \in \mathbb{R}^{N\times N}$ is nonnegative and irreducible and all the diagonal entries of A are positive, then $A^{N-1} > 0$ [44]. The following lemma from [16] is a generalization of this fact.

4.2.9. LEMMA. *[16] Let $A_i \in \mathbb{R}^{N\times N}$, $i = 1, \ldots, n$, be n nonnegative irreducible matrices with positive diagonal entries. If $k \geqslant N-1$ and $1 \leqslant i_1, \ldots, i_k \leqslant n$, then $A_{i_k} \cdots A_{i_2} A_{i_1} > 0$.*

4.2.10. LEMMA. *Let $\mathcal{P} = \{P_1, \ldots, P_n\}$ be a finite set of matrices and let $\mathcal{W} = \{W_1, \ldots, W_n\}$ be a set of stochastic irreducible matrices with positive diagonal entries. Assume that $\rho(P_i) < 1$, $|P_i| \leqslant W_i$, $i = 1, \ldots, n$, and if $|P_i| = W_i$, then the diagonal entries of P_i are all positive. Then for each sequence of matrices $P(1), P(2), P(3), \ldots$ from $\mathcal{P}$, the product $P(k) \cdots P(2)P(1)$ converges to zero as k goes to infinity.*

Proof. Consider a sequence of matrices $P(1), P(2), P(3), \ldots$ from $\mathcal{P}$. Let $Q = P(N-1) \cdots P(2)P(1)$. Lemma 4.2.2 implies that

$$|Q| = |P(N-1) \cdots P(2)P(1)| \leqslant |P(N-1)| \cdots |P(2)||P(1)| \leqslant W(N-1) \cdots W(2)W(1), \tag{4.16}$$

where $|P(i)| \leqslant W(i)$ and $W(i) \in \mathcal{W}$. Let $S = W(N-1) \cdots W(2)W(1)$. Since $W(i)$, $i = 1, \ldots, N-1$, are irreducible and they have positive diagonal entries, it follows from Lemma 4.2.9 that $S > 0$. If $|Q| \neq S$, then mimicking the proof in Lemma 4.2.6, one can show that $\rho(Q) < \rho(S) = 1$.

If $|Q| = S$, then it should hold that $W(N-1) \cdots W(i) \cdots W(1) = W(N-1) \cdots |P(i)| \cdots W(1) = |P(N-1)| \cdots |P(2)||P(1)|$ for all $i = 1, \ldots, N-1$ from (4.16). If for some i, $|P(i)| \neq W(i)$, then there exists some element, say the kjth element, of $W(i) - |P(i)|$ is nonzero. Since the diagonal elements of $W(l)$, $l = 1, \ldots, N-1$, are all positive, the ijth element of $W(N-1) \cdots (W(i) - |P(i)|) \cdots W(1)$ is also nonzero, which cannot happen. Thus one has that $|P(i)| = W(i)$, for all $i = 1, \ldots, N-1$. From the assumption of the lemma, the diagonal elements of $P(i)$ are all positive.

$P(i)$ satisfy the condition (4.8). Combining with the fact that $\rho(P(i)) < 1$ and $P(i)$ are irreducible, one has that $P(i)$ is structurally unbalanced. Thus there does not exist a diagonal matrix U satisfying $U^2 = I$ such that $UP(i)U \geqslant 0$. The conditions in Lemma 4.2.7 are satisfied and thus $\rho(Q) < 1$.

Since

$$\begin{aligned} |q_{ii}| &= \left| \sum_{k_1,\ldots,k_{N-2}=1,\ldots,N} (P(N-1))_{ik_{N-2}} \cdots (P(2))_{k_2k_1} (P(1))_{k_1 i} \right| \\ &= \sum_{k_1,\ldots,k_{N-2}=1,\ldots,N} \left| (P(N-1))_{ik_{N-2}} \cdots (P(2))_{k_2k_1} (P(1))_{k_1 i} \right|, \end{aligned}$$

it is true that $(P(N-1))_{ik_{N-2}} \cdots (P(2))_{k_2k_1}(P(1))_{k_1 i} \geqslant 0$ for all $k_1, \ldots, k_{N-2} = 1, \ldots, N$ or $(P(N-1))_{ik_{N-2}} \cdots (P(2))_{k_2k_1}(P(1))_{k_1 i} \leqslant 0$ for all $k_1, \ldots, k_{N-2} = 1, \ldots, N$. Since $(P(N-1))_{ii} \cdots (P(2))_{ii}(P(1))_{ii} > 0$, we have that $q_{ii} > 0,\ i = 1, \ldots, N$.

Similarly, $\rho(P(2N-2) \cdots P(N+1)P(N)) < 1$ and if $|P(2N-2) \cdots P(N+1)P(N)| = W(2N-2) \cdots W(N+1)W(N)$, then the diagonal elements of $P(2N-2) \cdots P(N+1)P(N)$ are all positive. Let $x(1) \in \mathbb{R}^N$ be an arbitrary vector and $x(i+1) = P(i)x(i),\ i = 1, 2, \ldots$. From Lemma 4.2.8, there exists a constant $0 \leqslant \tau < 1$ such that

$$||x(2N-1)|| = ||P(2N-2) \cdots P(N)P(N-1) \cdots P(1)x(1)|| \leqslant \tau ||x(1)||.$$

Note that the above arguments are valid for any matrix product $P(2N-2) \cdots P(2)$ $P(1)$ with $P(i) \in \mathcal{P},\ i = 1, \ldots, N-2$. Since the number of the matrices obtained by multiplying $2N-2$ matrices from $\mathcal{P}$ together is finite, there exists a uniform constant, still denoted by τ, $0 \leqslant \tau < 1$, such that for any vector $x(1)$, $||x(2N-1)|| \leqslant \tau ||x(1)||$, where $x(2N-1) = P(2N-2) \cdots P(2)P(1)x(1)$ and $P(i) \in \mathcal{P},\ i = 1, \ldots, N-2$.

Consider again a specific sequence of matrices $P(1), P(2), P(3), \ldots$ from $\mathcal{P}$. For $k \geqslant 1$, let $k = a(N-1) + b$, where $a,\ b$ are integers satisfying $a \geqslant 0,\ 0 \leqslant b < N-1$. We have that

$$||x(k)|| \leqslant ||x(a(N-1)+1)|| \leqslant \tau^a ||x(1)||. \tag{4.17}$$

Thus $\lim_{k\to\infty} ||x(k)|| = 0$ for any vector $x(1)$. This is equivalent to the conclusion that

$$\lim_{k\to\infty} P(k) \cdots P(2)P(1) = 0.$$

□

It is known that for a general time-varying system (4.2), the condition $\rho(P(t)) < 1$ cannot guarantee the asymptotic stability of system (4.2). Lemma 4.2.10 has constructed a sufficient condition to guarantee the asymptotic stability of system

(4.2), when $P(t)$ are picked from a finite set. The condition that if $|P_i| = W_i$, then the diagonal elements of P_i are positive is critical, and it cannot be removed. We provide an example to show that if it is not satisfied, the system can be unstable.

4.2.2. Example. Let

$$P(t) = \frac{1}{3}\begin{bmatrix} 1 & -1 & -1 \\ -1 & 1 & 1 \\ 1 & -1 & -1 \end{bmatrix}, \text{ if } t \text{ is odd; } P(t) = \frac{1}{3}\begin{bmatrix} 1 & -1 & 1 \\ -1 & 1 & -1 \\ -1 & 1 & -1 \end{bmatrix}, \text{ if } t \text{ is even.}$$

$|P(t)|$ are stochastic irreducible matrices and $\rho(P(t)) = \frac{1}{3}$. The conditions in Lemma 4.2.10 are satisfied except that some diagonal elements of $P(t)$ are negative. It is easy to check that

$$P(k)\cdots P(2)P(1) = \begin{cases} \frac{1}{3}\begin{bmatrix} 1 & -1 & -1 \\ -1 & 1 & 1 \\ -1 & 1 & 1 \end{bmatrix}, & k \text{ is even,} \\ P(1), & k \text{ is odd.} \end{cases} \tag{4.18}$$

The matrix product $P(k)\cdots P(2)P(1)$ does not converge as k goes to infinity and thus the system is unstable. □

Proof of Theorem 4.2.3. Pick an integer T such that T is a uniform bound for the length of time intervals $[t_i, t_{i+1}),\ i = 0, 1, 2, \ldots$. Since $P(t) \in \mathcal{P}$, $|P(t_{i+1} - 1)| \cdots |P(t_i + 1)||P(t_i)|$ are stochastic matrices with positive diagonal elements. Since $\cup_{t\in[t_i,t_{i+1})}\mathbb{G}(P(t))$ are strongly connected, the stochastic matrices $|P(t_{i+1} - 1)| \cdots |P(t_i + 1)||P(t_i)|$ are all irreducible [47]. Combining with the condition that there does not exist a diagonal matrix U such that $UP(t)U \geqslant 0,\ t_i \leqslant t < t_{i+1}$, it follows from Lemma 4.2.7 that $\rho(P(t_{i+1}-1)\cdots P(t_i+1)P(t_i)) < 1$. If $|P(t_{i+1}-1)\cdots P(t_i+1)P(t_i)| = |P(t_{i+1} - 1)| \cdots |P(t_i + 1)||P(t_i)|$, then the diagonal elements of $P(t_{i+1} - 1)\cdots P(t_i+1)P(t_i)$ are all positive, since the diagonal elements of $P(t)$ are all positive. The number of the matrices obtained by multiplying no more than T matrices from a finite set is also finite. Thus $P(t_{i+1}-1)\cdots P(t_i+1)P(t_i),\ i = 0, 1, \ldots$ is a sequence of matrices from a finite set. The conditions in Lemma 4.2.10 are satisfied and we conclude that $\lim_{t\to\infty} P(t)\cdots P(2)P(1) = 0$, which completes the proof. □

A common Lyapunov function can be found for the time-varying system (4.2) to show its asymptotic convergence in the spirit of Lemmas 4.2.8 and 4.2.10. Let $V(x(t)) = ||x(t)||$. Then $||x(t+1)|| \leqslant ||x(t)||$ and $V(x(t))$ is a non-increasing function along the solution of system (4.2). Further

$$||x((k+1)(2N-2))|| \leqslant \tau^{k+1}||x(0)||,\ k = 0, 1, 2, \ldots,$$

for some constant $0 \leqslant \tau < 1$, meaning that $V(x(t))$ will decrease strictly after updating at most $2N - 2$ steps. This establishes the strictly decreasing property of $V(x(t))$, proving the asymptotic convergence of system (4.2). The convergence speed can also be characterized.

4.3 Distributed continuous-time algorithms

4.3.1 Continuous-time updates under fixed topologies

We first consider the continuous-time algorithm (4.4) under fixed topologies. Let $\mathbb{G}$ be a signed graph representing the interaction graph topology, let $A \in \mathbb{R}^{N\times N}$ be the signed adjacency matrix, and let $L = (l_{ij})_{N\times N}$ be the signed Laplacian matrix given by (2.3). System (4.4) and system (4.5) become

$$\dot{x}_i = -\sum_{j=1}^{N} |a_{ij}|(x_i - sgn(a_{ij})x_j), \qquad i = 1, \ldots, N, \tag{4.19}$$

and

$$\dot{x} = -Lx. \tag{4.20}$$

4.3.1. Lemma. *A strongly connected signed graph $\mathbb{G}$ with the signed adjacency matrix A is structurally balanced if and only if one of the following equivalent conditions holds:*
(a) All cycles of $\mathbb{G}$ are positive;
(b) There exists a unique diagonal matrix U satisfying $U^2 = I$ such that UAU is nonnegative;
(c) 0 is an eigenvalue of the signed Laplacian matrix L.

Proof. Note that if $\mathbb{G}$ is structurally balanced, then $a_{ij}a_{ji} \geqslant 0,\ i, j = 1, \ldots, N$. (a) and (b) follow from Lemma 2 in [3] and (b)$\Longrightarrow$(c) is clear. We only prove (c)$\Longrightarrow$ $\mathbb{G}$ is structurally balanced.

Assume that 0 is an eigenvalue of L. Since $l_{ii} = \sum_{j=1,j\neq i}^{N} |a_{ij}|$, from Lemma 2.2.1, it follows that 0 is a boundary point of the Gersgorin region $G(L)$. Combining with the fact that L is irreducible, from Lemma 4.2.1, there exists an eigenvector x of 0 satisfying $|x_i| = |x_j| > 0,\ i, j = 1, \ldots, N$. From $Lx = 0$, one has

$l_{ii}x_i + \sum_{j=1,j\neq i}^{N} l_{ij}x_j = 0$, that is $\sum_{j=1,j\neq i}^{N} |a_{ij}|x_i = \sum_{j=1,j\neq i}^{N} a_{ij}x_j$. In addition, the following inequality holds

$$\sum_{j=1,j\neq i}^{N} a_{ij}x_j \leqslant \left|\sum_{j=1,j\neq i}^{N} a_{ij}x_j\right| \leqslant \sum_{j=1,j\neq i}^{N} |a_{ij}x_j| = \sum_{j=1,j\neq i}^{N} |a_{ij}||x_i|. \tag{4.21}$$

This implies that $x_i a_{ij} x_j \geqslant 0$ for all $j \neq i,\ i,j = 1,\ldots,N$. Mimicking the proof in Lemma 4.2.5(c), one can prove that $\mathbb{G}$ is structurally balanced. □

Note that in [3], it is assumed that the weights in the graph satisfy the *digon sign-symmetric* assumption, which requires that $a_{ij}a_{ji} \geqslant 0,\ i,j = 1,\ldots,N$. This assumption obviously holds if $\mathbb{G}$ is structurally balanced. Under this assumption, the equivalent conditions (a) (b) (c) to structural balance have been established in [3]. The condition (c) is derived by exploring the property of the graph $\mathbb{G}(A^s)$. Here without looking at the property of the graph $\mathbb{G}(A^s)$, we are enabled to remove this assumption and directly show the equivalence between (c) and the structural balance of $\mathbb{G}$. Note that Lemma 2(1) in [3], which states that $\mathbb{G}(A^s)$ is structurally balanced, does not hold without the digon sign-symmetric assumption. Only one part of the necessary and sufficient condition is valid. The correct part is that $\mathbb{G}(A^s)$ is structurally balanced if $\mathbb{G}(A)$ is structurally balanced. The converse is not valid, which can be illustrated by a counterexample. Let

$$A = \begin{bmatrix} 0 & 1 & 1 \\ 1 & 0 & 1 \\ -1 & 1 & 0 \end{bmatrix},\ A^s = \begin{bmatrix} 0 & 1 & 0 \\ 1 & 0 & 1 \\ 0 & 1 & 0 \end{bmatrix}.$$

$\mathbb{G}(A^s)$ is structurally balanced, while $\mathbb{G}(A)$ is not.

4.3.1. Corollary. *A strongly connected signed graph $\mathbb{G}$ with the signed adjacency matrix A is structurally unbalanced if and only if one of the following equivalent conditions holds:*
(a) $\mathbb{G}$ has at least one negative cycle;
(b) There does not exist a diagonal matrix U satisfying $U^2 = I$ such that UAU is nonnegative;
(c) $Re(\lambda_i(L)) < 0,\ i = 1,\ldots,N$.

The following result can be straightforwardly derived by making a transformation to the system state of (4.20).

4.3.1. Theorem. *Let $\mathbb{G}$ be a strongly connected signed graph. System (4.20) admits a polarization if and only if $\mathbb{G}$ is structurally balanced. Furthermore, if U is a diagonal matrix satisfying $U^2 = I$ such that UAU is nonnegative, then the state of* *system (4.20) asymptotically converges to* $\lim_{t\to\infty} x(t) = v^T U x(0) U \mathbf{1}$*, where* v *is a left normalized eigenvector of* ULU *corresponding to 0 such that* $v^T \mathbf{1} = 1$*. If* $\mathbb{G}$ *is structurally unbalanced, then* $\lim_{t\to\infty} x(t) = 0$ *for every initial value.*

4.3.2 Continuous-time updates under time-varying topologies

In this section, we consider the case when the interaction graph topologies are dynamically changing. Assume that at time t the signed adjacency matrix is $A(t)$ and the signed Laplacian matrix is $L(t)$. Assume that $A(t)$ and $L(t)$ are piecewise constant functions and the interaction graph topologies change at time instants $t_1, t_2, \ldots$. System (4.5) can be rewritten as

$$\dot{x}(t) = -L(t_i)x(t), \quad t \in [t_i, t_i + \tau_i), \tag{4.22}$$

where $t_0 = 0$ is the initial time, and $\tau_i = t_{i+1} - t_i,\ i = 0, 1, \ldots$ are the dwell time. Assume that $\tau_i \geqslant \tau_T$ for all $i = 0, 1, \ldots$.

Given a signed adjacency matrix A, if there is a diagonal matrix U satisfying $U^2 = I$ such that UAU is nonnegative, then $\bar{L} = ULU$ is a Laplacian matrix that has nonnegative off-diagonal elements. Let τ be a positive number, let $P = e^{-L\tau}$ and let $\bar{P} = e^{-\bar{L}\tau}$. One has

$$\bar{P} = e^{-\bar{L}\tau} = e^{-ULU\tau} = Ue^{-L\tau}U = UPU. \tag{4.23}$$

$\bar{P} = e^{-\bar{L}\tau}$ is a stochastic matrix with positive diagonal entries [81]. We conclude that $p_{ii} > 0$ and $\sum_{j=1}^{N} |p_{ij}| = 1,\ i = 1, \ldots, N$. P satisfies the condition (4.8). If a strongly connected signed graph $\mathbb{G}(A)$ is structurally balanced, then there is a diagonal matrix U satisfying $U^2 = I$ such that UAU is nonnegative. Thus $P = e^{-L\tau}$ satisfies the condition (4.8), P is irreducible, and it is structurally balanced.

Let $\mathcal{A} = \{A_1, A_2, \ldots, A_n\}$ be a finite set of signed adjacency matrices.

4.3.2. Theorem. *Assume that* $A(t) \in \mathcal{A}$ *is piecewise constant, and there is a diagonal matrix* U *such that* $UA(t)U \geqslant 0$ *for all* $t = 0, 1, 2, \ldots$*. Assume that there exists an infinite sequence of contiguous, nonempty, uniformly bounded time intervals* $[t_{i_k}, t_{i_{k+1}})$*,* $k \geqslant 0$*, starting at* $t_{i_0} = 0$ *with the property that across each time interval* $[t_{i_k}, t_{i_{k+1}})$*, the union of the graphs are strongly connected. Then system (4.22) admits a polarization and* $\lim_{t\to\infty} x(t) = \alpha U \mathbf{1}$*, where* α *is some constant.*

4.3.3. Theorem. *Assume that the signed adjacency matrices in* $\mathcal{A}$ *are all irreducible and* $A(t) \in \mathcal{A}$ *is piecewise constant. There exists an infinite sequence of contiguous, nonempty, uniformly bounded time intervals* $[t_{i_k}, t_{i_{k+1}})$*,* $k \geqslant 0$*, starting at* $t_{i_0} = 0$ *with the property that there does not exist a diagonal matrix* U *satisfying* $U^2 = I$ *such that* $UA(t)U \geqslant 0,\ t_{i_k} \leqslant t < t_{i_{k+1}}$*. Then system (4.22) converges to zero asymptotically.*

Proof. The solution to (4.22) is given by

$$x(t) = e^{-L(t_k)(t-t_k)} \cdots e^{-L(t_1)\tau_1} e^{-L(t_0)\tau_0} x(0),$$

where k is the largest nonnegative integer satisfying $t_k < t$. Let $P_i = e^{-L(t_i)\tau_i}$, $i = 0, 1, \ldots$. P_i satisfy the condition (4.8), and P_i are irreducible and structurally balanced. The rest of the proof is similar to the argument used in Theorem 4.2.3. $\square$

4.4 Illustrative example

In this section, we perform simulation studies on discrete-time model (4.2). Let P_1, P_2, and P_3 be given by

$$P_1 = \left[\begin{array}{ccccc:ccccc}
\frac{1}{3} & \frac{1}{3} & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -\frac{1}{3} \\
\frac{1}{4} & \frac{1}{4} & \frac{1}{4} & 0 & 0 & 0 & 0 & 0 & -\frac{1}{4} & 0 \\
0 & \frac{1}{3} & \frac{1}{3} & \frac{1}{3} & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & \frac{1}{4} & \frac{1}{4} & \frac{1}{4} & 0 & -\frac{1}{4} & 0 & 0 & 0 \\
0 & 0 & 0 & \frac{1}{4} & \frac{1}{4} & -\frac{1}{4} & 0 & -\frac{1}{4} & 0 & 0 \\
\hdashline
0 & 0 & 0 & 0 & -\frac{1}{3} & \frac{1}{3} & \frac{1}{3} & 0 & 0 & 0 \\
0 & 0 & 0 & -\frac{1}{4} & 0 & \frac{1}{4} & \frac{1}{4} & \frac{1}{4} & 0 & 0 \\
0 & 0 & 0 & 0 & -\frac{1}{4} & 0 & \frac{1}{4} & \frac{1}{4} & \frac{1}{4} & 0 \\
0 & -\frac{1}{3} & 0 & 0 & 0 & 0 & 0 & \frac{1}{3} & \frac{1}{3} & 0 \\
-\frac{1}{2} & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & \frac{1}{2}
\end{array}\right],$$

$$P_2 = \left[\begin{array}{ccccc:ccccc}
\frac{1}{2} & \frac{1}{2} & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
\frac{1}{5} & \frac{1}{5} & \frac{1}{5} & 0 & \frac{1}{5} & 0 & 0 & 0 & 0 & -\frac{1}{5} \\
0 & \frac{1}{4} & \frac{1}{4} & \frac{1}{4} & 0 & -\frac{1}{4} & 0 & 0 & 0 & 0 \\
0 & 0 & \frac{1}{3} & \frac{1}{3} & 0 & 0 & -\frac{1}{3} & 0 & 0 & 0 \\
0 & \frac{1}{4} & 0 & 0 & \frac{1}{4} & 0 & 0 & -\frac{1}{4} & -\frac{1}{4} & 0 \\
\hdashline
0 & 0 & -\frac{1}{3} & 0 & 0 & \frac{1}{3} & \frac{1}{3} & 0 & 0 & 0 \\
0 & 0 & 0 & -\frac{1}{4} & 0 & \frac{1}{4} & \frac{1}{4} & \frac{1}{4} & 0 & 0 \\
0 & 0 & 0 & 0 & -\frac{1}{4} & 0 & \frac{1}{4} & \frac{1}{4} & \frac{1}{4} & 0 \\
0 & 0 & 0 & 0 & -\frac{1}{3} & 0 & 0 & \frac{1}{3} & \frac{1}{3} & 0 \\
0 & -\frac{1}{2} & 0 & 0 & 0 & 0 & 0 & 0 & 0 & \frac{1}{2}
\end{array}\right],$$

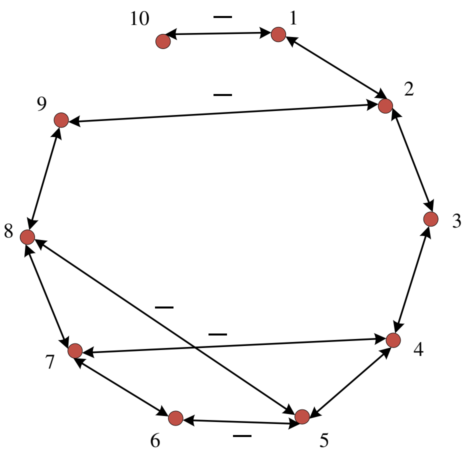


Figure 4.2: *The graph $\mathbb{G}(P_1)$ associated to P_1.*

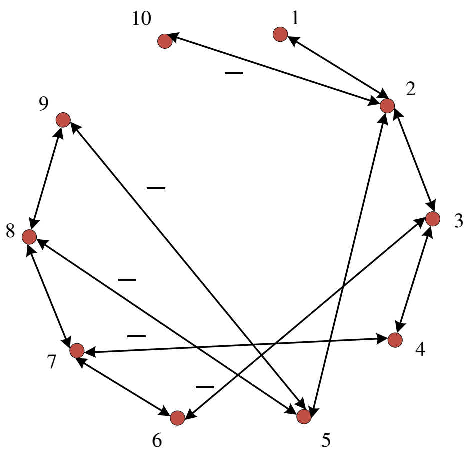


Figure 4.3: *The graph $\mathbb{G}(P_2)$ associated to P_2.*

$$P_3 = \left[\begin{array}{ccccc:cccc:c}
\frac{1}{3} & \frac{1}{3} & 0 & 0 & 0 & 0 & 0 & 0 & 0 & \frac{1}{3} \\
\frac{1}{4} & \frac{1}{4} & \frac{1}{4} & 0 & 0 & 0 & 0 & 0 & -\frac{1}{4} & 0 \\
0 & \frac{1}{3} & \frac{1}{3} & \frac{1}{3} & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & \frac{1}{4} & \frac{1}{4} & \frac{1}{4} & 0 & -\frac{1}{4} & 0 & 0 & 0 \\
0 & 0 & 0 & \frac{1}{4} & \frac{1}{4} & -\frac{1}{4} & 0 & -\frac{1}{4} & 0 & 0 \\
\hdashline
0 & 0 & 0 & 0 & -\frac{1}{3} & \frac{1}{3} & \frac{1}{3} & 0 & 0 & 0 \\
0 & 0 & 0 & -\frac{1}{4} & 0 & \frac{1}{4} & \frac{1}{4} & \frac{1}{4} & 0 & 0 \\
0 & 0 & 0 & 0 & -\frac{1}{4} & 0 & \frac{1}{4} & \frac{1}{4} & \frac{1}{4} & 0 \\
0 & -\frac{1}{3} & 0 & 0 & 0 & 0 & 0 & \frac{1}{3} & \frac{1}{3} & 0 \\
\hdashline
\frac{1}{2} & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & \frac{1}{2}
\end{array}\right].$$

The graphs associated to P_1, P_2, and P_3 are denoted by $\mathbb{G}(P_1), \mathbb{G}(P_2)$, and $\mathbb{G}(P_3)$ and they are illustrated in Figs 4.2-4.4. The edges with negative weights are labeled by "$-$" signs and the rest edges are with positive weights. It is easy to see that $\mathbb{G}(P_1), \mathbb{G}(P_2)$, and $\mathbb{G}(P_3)$ are all structurally balanced. The bipartitions of the node set $\{v_1, \ldots, v_{10}\}$ for $\mathbb{G}(P_1)$ and $\mathbb{G}(P_2)$ satisfying Lemma 4.2.5(a) are the same, which is $\{\{v_1, \ldots, v_5\}, \{v_6, \ldots, v_{10}\}\}$, and hence $U = diag\{1, 1, 1, 1, 1, -1, -1, -1, -1, -1\}$ is a diagonal matrix satisfying that $U^2 = I$ and $UP_1U \geqslant 0$, $UP_2U \geqslant 0$. $\mathbb{G}(P_3)$ has a different bipartition satisfying Lemma 4.2.5(a), which is $\{\{v_1, \ldots, v_5, v_{10}\}, \{v_6, v_7, v_8, v_9\}\}$, and hence $\bar{U} = diag\{1, 1, 1, 1, 1, -1, -1, -1, -1, 1\}$ is a diagonal matrix satisfying that $\bar{U}^2 = I$ and $\bar{U}P_3\bar{U} \geqslant 0$. In the simulations, take the initial values of the ten agents to be $x(0) = [4, -2, 3, -3, 5, -2, -3, -2, 1, 0]^T$.

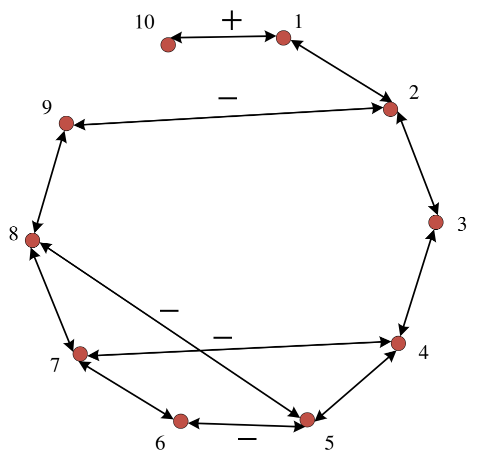


Figure 4.4: *The graph $\mathbb{G}(P_3)$ associated to P_3.*

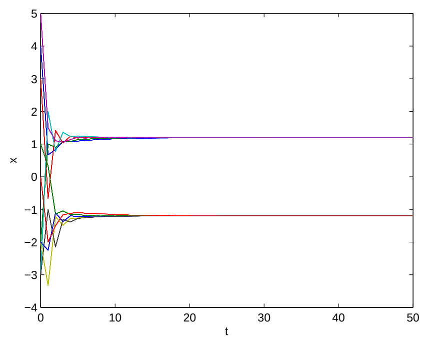


Figure 4.5: *The evolution of the agents' states when the graph topologies switch between $\mathbb{G}(P_1)$ with $\mathbb{G}(P_2)$.*

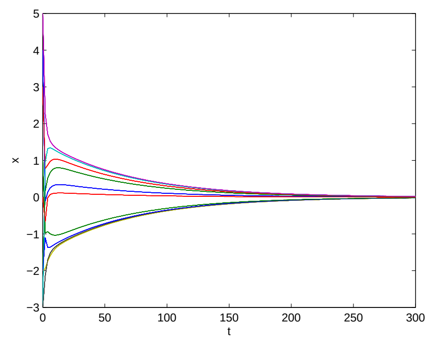


Figure 4.6: *The evolution of the agents' states when the graph topologies switch between $\mathbb{G}(P_1)$ with $\mathbb{G}(P_3)$.*

First consider the switched system (4.2) with

$$P(t) = \begin{cases} P_1, & t \text{ is even}, \\ P_2, & t \text{ is odd}. \end{cases} \tag{4.24}$$

Then the states of all the agents evolve to two opposite values since $\mathbb{G}(P_1)$ and $\mathbb{G}(P_2)$

share the same bipartition of the node set, which is illustrated in Fig. 4.5. If

$$P(t) = \begin{cases} P_1, \ t \text{ is even}, \\ P_3, \ t \text{ is odd}, \end{cases} \tag{4.25}$$

the conditions in Theorem 4.2.3 are satisfied. We can conclude that the states of all the agents asymptotically converge to zero, which is also verified by Fig. 4.6. Since $\mathbb{G}(P_1)$ is structurally balanced, for system (4.2) with static interaction topology $P(t) = P_1, \ t = 0, 1, \ldots$, the states of the agents will converge to two opposite values. When we compare $\mathbb{G}(P_1)$ with $\mathbb{G}(P_3)$, it is observed that only the weight of the edge between v_1 and v_{10} changes the sign. However, small variations of the interaction graph topology has led to significant changes in the dynamical behavior of the system.

4.5 Conclusion

In this chapter, we have generalized the distributed algorithms introduced in Chapter 2 to more general settings where the couplings between pairs of agents in a network could be positive or negative. Both discrete-time and continuous-time algorithms have been discussed under dynamically changing interaction topologies. By making use of the notion of structural balance, sufficient conditions have been constructed to guarantee that the states of the agents converge to two opposite values or converge to zero.

We have considered the setting that the interaction graph topologies are dynamically changing with time but not affected by the states of the agents. In realistic social networks, the relationship between two persons is likely to be affected by the variations of their opinions about a subject. It is of great interest to consider the case when the network topologies are time-varying and also dependent on the system's state.

Chapter 5

Distributed Clustering Algorithms

Various algorithms have been successfully constructed to cause all the agents in a group to converge to the same value asymptotically [47, 81, 16]. At the same time, there is a growing interest to study how an interconnected group may incorporate or evolve into different sub-groups called clusters. In contrast to the widely studied synchronization processes, in the cluster synchronization problem studied in this chapter, we require all the interconnected agents to evolve into several clusters and each agent only to synchronize within its cluster. We focus on the n-cluster synchronization problem to be defined in the next section. We present in this chapter three approaches that may lead to clustering behavior in connected networks consisting of locally interacting agents. The first approach is that agents have different self-dynamics, and those agents having the same self-dynamics may evolve into the same cluster. When the agents' self-dynamics are identical, we present two other approaches by which cluster synchronization might be achieved. One is the presence of delays and the other is the existence of both positive and negative couplings between the agents.

5.1 Problem formulation

In this chapter we aim to study n-cluster synchronization problem, in which a coupled multi-agent system is required to split into n clusters, so that the agents synchronize with one another in the same cluster, but differences exist between different clusters [105]. We first give a formal definition of n-cluster synchronization. As in previous chapters, directed weighted graphs are used to describe the couplings among the agents. Let $\mathbb{G}$ be a directed weighted graph representing the interaction topology among the agents and let A and L be the corresponding adjacency matrix and Laplacian matrix of $\mathbb{G}$. Consider the following extensively studied model in the synchronization study for a complex network [77, 51] that consists of N coupled agents

$$\begin{aligned}\dot{x}_i(t) &= f_i(t, x_i(t)) + c \sum_{j=1, j\neq i}^{N} a_{ij}\Gamma(x_j(t) - x_i(t)) \\ &= f_i(t, x_i(t)) - c \sum_{j=1}^{N} l_{ij}\Gamma x_j(t),\end{aligned} \tag{5.1}$$

where $x_i \in \mathbb{R}^m$ denotes the state of agent i, $i = 1, \ldots, N$, $f_i : [0, \infty) \times \mathbb{R}^m \to \mathbb{R}^m$ is continuous and globally Lipschitzian with Lipschitz constant K_i, namely

$$||f_i(t, \xi_1) - f_i(t, \xi_2)||_2 \leqslant K_i||\xi_1 - \xi_2||_2, \tag{5.2}$$

for all $(t, \xi_1),\ (t, \xi_2) \in [0, \infty) \times \mathbb{R}^m$, $c > 0$ is the coupling strength, a_{ij} is the ijth element of A, l_{ij} is the ijth element of L, and the diagonal matrix $\Gamma = diag\{\gamma_1, \ldots, \gamma_m\}$ denotes the inner coupling with $\gamma_k \geqslant 0$ for $k = 1, \ldots, m$. System (5.1) has a unique solution which exists for all $t \geqslant 0$ [33].

We say that $\{C_1, C_2, \ldots, C_n\}$, $n > 1$, is a *partition* of the set $\mathcal{N} = \{1, 2, \ldots, N\}$ if $C_i \neq \varnothing$, $C_i \bigcap C_j = \varnothing$ and $\bigcup_{i=1}^{n} C_i = \mathcal{N}$; furthermore, we use $\hat{i}$ to denote the index of that subset of the partition in which the number i lies, i.e., $i \in C_{\hat{i}}$. Obviously, $1 \leqslant \hat{i} \leqslant n$. We say that agents i and j are in the same cluster if $\hat{i} = \hat{j}$. Now we are ready to define what we mean by cluster synchronization.

5.1.1. Definition. *For a given initial condition $x(0) = [x_1^T(0), \ldots, x_N^T(0)]^T$, where $x_i(0) \in \mathbb{R}^m$, $1 \leqslant i \leqslant N$, system (5.1) is said to realize n-cluster synchronization with the partition $\{C_1, C_2, \ldots, C_n\}$ if $\lim_{t\to\infty} ||x_i(t) - x_j(t)|| = 0$ for $\hat{i} = \hat{j}$ and $\limsup_{t\to\infty} ||x_i(t) - x_j(t)|| > 0$ for $\hat{i} \neq \hat{j}$.*

5.1.1. Remark. *In [112], a similar concept called the "group consensus" of a multi-agent system is defined, which is weaker than the cluster synchronization defined here because we require in addition that the differences between different clusters do not go to 0 as time goes to infinity. A different type of clustering behavior is considered in [1, 91], where the differences between agents in the same cluster are bounded, while the differences between agents in different clusters grow unbounded as time goes to infinity.*

In the synchronization study literature, the f_i in (5.1) are often referred to as the *self-dynamics* of agent i. In what follows, we discuss clustering approaches according to whether the agents' self-dynamics are identical.

5.2 Clustering with different self-dynamics

We first illustrate how agents governed by different linear dynamics might evolve into different clusters. We consider the case when some agents are under constant forcings and the others are not. The dynamics of the former are

$$\dot{x}_i(t) = -x_i(t) + b_{\hat{i}} - \sum_{j=1}^{N} l_{ij} x_j(t) \tag{5.3}$$

where l_{ij} is the ijth element of the Laplacian matrix L satisfying $l_{ij} \leqslant 0$, for $i \neq j$, and $b_{\hat{i}}$ are constants with $b_{\hat{i}} \neq b_{\hat{j}}$ for $\hat{i} \neq \hat{j}$. The dynamics of the latter are

$$\dot{x}_i(t) = -\sum_{j=1}^{N} l_{ij} x_j(t). \tag{5.4}$$

Comparing (5.3) and (5.4) with (5.1), we have taken f_i to be affine functions, Γ an identity matrix, $c = 1$, and $m = 1$. The results derived in this section can be easily extended to the more general case when $c > 0$ and $m \geqslant 1$. Since the constant forcing terms sometimes come from the agents' knowledge about their preferred values, the agents described by (5.3) are called *informed agents* and naturally the agents described by (5.4) are called *naive agents* since they do not have prior knowledge and have to rely on the interactions with their peers to evolve. In the next two subsections, we provide some sufficient and/or necessary conditions for systems of informed and naive agents to converge to n clusters.

5.2.1 Systems of informed agents

In this subsection, we consider the case when the system only consists of N informed agents described by (5.3) for $1 \leqslant i \leqslant N$. Assume that we have labeled the agents in such a way that the first l_1 agents are under the forcing b_1, the next l_2 agents are under b_2, and so on. Then the system can be written in a compact form

$$\dot{x}(t) = -x(t) + \bar{b} - Lx(t) = -\bar{L}x(t) + \bar{b}, \tag{5.5}$$

where $x = [x_1, x_2, \ldots, x_N]^T \in \mathbb{R}^N$, $\bar{L} = L + I$, and $\bar{b} = [b_1 \mathbf{1}_{l_1}^T, \ldots, b_{n-1}\mathbf{1}_{l_{n-1}}^T, b_n \mathbf{1}_{l_n}^T]^T$ with $l_1 + \cdots + l_n = N$.

We further write the Laplacian matrix L in the following block matrix form:

$$L = \begin{bmatrix} L_{11} & L_{12} & \cdots & L_{1n} \\ L_{21} & L_{22} & \cdots & L_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ L_{n1} & L_{n2} & \cdots & L_{nn} \end{bmatrix},$$

where $L_{ij} \in \mathbb{R}^{l_i \times l_j}$, $1 \leqslant i, j \leqslant n$. Since the row sums of L are zero, we know the row sums of $-\bar{L}$ are -1. In addition, $-\bar{L}$ has nonnegative off-diagonal elements. Hence, $-\bar{L}$ is invertible and the eigenvalues of $-\bar{L}$ are all located in the open left-half plane. The equilibrium of system (5.5) is $x^* = \bar{L}^{-1}\bar{b}$. Define $y(t) = x(t) - x^*$; then $\dot{y}(t) = -\bar{L}y(t)$. It is obvious that $y(t) \to 0$ as $t \to \infty$. Thus x^* is a globally stable equilibrium of system (5.5). In fact, we can say more about the structures of x^* as follows.

5.2.1. Theorem. *For any initial condition, system (5.5) of informed agents achieves n-cluster synchronization for almost all (in the sense of Lebesgue measure) b_i, $1 \leqslant i \leqslant n$, with $b_i \neq b_j$ for $i \neq j$, if the block matrices L_{ij}, $1 \leqslant i, j \leqslant n$ and $i \neq j$, have constant row sums.*

The proof of this theorem makes use of the following lemma.

5.2.1. Lemma. *Consider the matrix $P = (P_{ij})_{N \times N}$ where $P_{ij} \in \mathbb{R}^{l_i \times l_j}$, $1 \leqslant i, j \leqslant n$. Suppose that P is invertible and that its inverse is $Q = (Q_{ij})_{N \times N}$, where Q is partitioned in the same way as P. If the matrices P_{ij} have constant row sums for $1 \leqslant i, j \leqslant n$, then the matrices Q_{ij} also have constant row sums for $1 \leqslant i, j \leqslant n$. In addition, let r_{ij} denote the row sum of P_{ij} and s_{ij} denote that of Q_{ij}; then $RS = I_{n \times n}$, where $R = (r_{ij})_{n \times n}$ and $S = (s_{ij})_{n \times n}$.*

Proof: From $QP = I$, one has

$$\sum_{k=1}^{n} Q_{1k}P_{kj} = \begin{cases} I, & j = 1, \\ O, & j \neq 1, \end{cases}$$

Since P_{kj} have constant row sums r_{kj}, summing up the elements in each row of P_{kj} gives

$$\left(\begin{bmatrix} r_{11} & r_{21} & \cdots & r_{n1} \\ r_{12} & r_{22} & \cdots & r_{n2} \\ \vdots & \vdots & \ddots & \vdots \\ r_{1n} & r_{2n} & \cdots & r_{nn} \end{bmatrix} \otimes I \right) \begin{bmatrix} Q_{11}\mathbf{1} \\ Q_{12}\mathbf{1} \\ \vdots \\ Q_{1n}\mathbf{1} \end{bmatrix} = \begin{bmatrix} \mathbf{1} \\ \mathbf{0} \\ \vdots \\ \mathbf{0} \end{bmatrix}. \tag{5.6}$$

Since P is invertible, so is R. Combining with (5.6), we know that Q_{1j} have constant row sums for $1 \leqslant j \leqslant n$. In addition, the row sums s_{1j} of Q_{1j} satisfy

$$[s_{11}, s_{12}, \ldots, s_{1n}]^T = (R^T)^{-1}[1, 0, \ldots, 0]^T.$$

Using a similar calculation, it is easy to check that all Q_{ij} have constant row sums for $1 \leqslant i, j \leqslant n$, and $S^T = R^{-T}I$; that is $SR = I$. □

Now we are ready to prove Theorem 5.2.1.

Proof of Theorem 5.2.1: Let $Q = (Q_{ij})_{N\times N}$ be the inverse of $-\bar{L}$. Since $\bar{L}_{ij}$, $i \neq j$, have constant row sums and the row sums of $-\bar{L}$ are -1, it follows from Lemma 5.2.1 that Q_{ij} have constant row sums for $1 \leqslant i, j \leqslant n$. Denote the row sum of $-\bar{L}_{ij}$ by r_{ij} and that of Q_{ij} by s_{ij}. Then again from Lemma 5.2.1, we know that $S = R^{-1}$, where $R = (r_{ij})_{n\times n}$, and $S = (s_{ij})_{n\times n}$. So all the agents in the ith cluster have the same asymptotic value $-\sum_{j=1}^{n} s_{ij} b_j$.

Next we show that all the b_i that do not lead to n-cluster synchronization come from a set which has zero Lebesgue measure. Let $\mathcal{S} = \{x = [x_1, \ldots, x_n]^T \in \mathbb{R}^n : \ x_i = x_j$ for some $i \neq j$ with $1 \leqslant i, j \leqslant n\}$, and let the smooth linear map $g : \mathbb{R}^n \to \mathbb{R}^n$ be defined by $g(x) = Rx$. Then it is easy to check that $\mathcal{S}$ has zero Lebesgue measure; so does $g(\mathcal{S})$. Let

$$\begin{aligned} \mathcal{U} = \{b = [b_1, \ldots, b_n]^T \in \mathbb{R}^n : \quad & b_i \neq b_j \text{ for } i \neq j; \\ & (R^{-1}b)_i = (R^{-1}b)_j \text{ for some } i \neq j \text{ and } 1 \leqslant i, j \leqslant n\}. \end{aligned}$$

One has $\mathcal{U} \subset g(\mathcal{S})$, which implies that $\mathcal{U}$ has zero Lebesgue measure. If $b \notin \mathcal{U}$, system (5.5) realizes n-cluster synchronization, which completes the proof. □

The condition given in Theorem 5.2.1 is a sufficient condition and it may not be necessary when $n > 2$. However, for the special case when $n = 2$, the condition is also necessary as shown in the following result.

5.2.2. THEOREM. *System (5.5) under any pair of distinct forcings $b_1 \neq b_2$ achieves 2-cluster synchronization for any initial condition if and only if the block matrices $L_{ij}, 1 \leqslant i, j \leqslant 2$ and $i \neq j$, have constant row sums.*

Proof: (Sufficiency) Let $Q = \begin{bmatrix} Q_{11} & Q_{12} \\ Q_{21} & Q_{22} \end{bmatrix}_{N\times N}$ be the inverse of $-\bar{L}$. It follows from the fact that $-\bar{L}_{ij}$ have constant row sums r_{ij} and Lemma 5.2.1 that Q_{ij} have constant row sums s_{ij} and

$$S = \begin{bmatrix} -\frac{r_{21}+1}{r_{12}+r_{21}+1} & -\frac{r_{12}}{r_{12}+r_{21}+1} \\ -\frac{r_{21}}{r_{12}+r_{21}+1} & -\frac{r_{12}+1}{r_{12}+r_{21}+1} \end{bmatrix}.$$

Thus solutions of system (5.5) converge to

$$x^* = \bar{L}^{-1}\bar{b} = -\begin{bmatrix} (b_1 s_{11} + b_2 s_{12})\mathbf{1}_{l_1} \\ (b_1 s_{21} + b_2 s_{22})\mathbf{1}_{l_2} \end{bmatrix}.$$

It is easy to check that $-b_1 s_{11} - b_2 s_{12} \neq -b_1 s_{21} - b_2 s_{22}$ since $b_1 \neq b_2$. Thus 2-cluster synchronization has been realized.

(Necessity) Suppose that system (5.5) realizes 2-cluster synchronization with final values $\bar{x}_1$ and $\bar{x}_2$. Let $\mathcal{N}_1 = \{k \in \mathcal{N}$; the final value of $x_k(t)$ is $\bar{x}_1\}$. We first show that every agent under the same constant forcing is in the same cluster. Suppose on the contrary that the ith and jth agents both under constant forcing b_1 have different final values $\bar{x}_1$ and $\bar{x}_2$; then one has

$$\begin{aligned} 0 &= -\bar{x}_1 + b_1 - \sum_{k \in \mathcal{N}/\mathcal{N}_1, k \neq i} l_{ik}(\bar{x}_2 - \bar{x}_1), \\ 0 &= -\bar{x}_2 + b_1 - \sum_{k \in \mathcal{N}_1, k \neq j} l_{jk}(\bar{x}_1 - \bar{x}_2). \end{aligned}$$

It follows that $(\bar{x}_2 - \bar{x}_1)(1 - \sum_{k \in \mathcal{N}/\mathcal{N}_1, k \neq i} l_{ik} - \sum_{k \in \mathcal{N}_1, k \neq j} l_{jk}) = 0$, which contradicts $\bar{x}_2 - \bar{x}_1 \neq 0$ and $1 - \sum_{k \in \mathcal{N}/\mathcal{N}_1, k \neq i} l_{ik} - \sum_{k \in \mathcal{N}_1, k \neq j} l_{jk} > 0$.

From the proof of sufficiency, we find that the equilibrium of system (5.5) is

$$x^* = -\begin{bmatrix} b_1 Q_{11} \mathbf{1}_{l_1} + b_2 Q_{12} \mathbf{1}_{l_2} \\ b_1 Q_{21} \mathbf{1}_{l_1} + b_2 Q_{22} \mathbf{1}_{l_2} \end{bmatrix}.$$

Let the ith row sums of Q_{11} and Q_{12} be t_{i1} and t_{i2} respectively. Then, for any $1 \leqslant i, j \leqslant l_1$ and $b_1 \neq b_2$, we have $-b_1 t_{i1} - b_2 t_{i2} = -b_1 t_{j1} - b_2 t_{j2}$. It follows that $t_{i1} = t_{j1}$ and $t_{i2} = t_{j2}$ for $1 \leqslant i, j \leqslant l_1$. Thus, Q_{11} and Q_{12} have constant row sums. Applying similar arguments to Q_{21} and Q_{22}, one can conclude that L_{12} and L_{21} have constant row sums in view of Lemma 5.2.1. □

In the next subsection, we consider the systems that consist of not only informed agents, but also naive agents.

5.2.2 Systems of informed and naive agents

Now consider the system consisting of $n-1$ clusters of informed agents and one cluster of naive agents, whose dynamics are described respectively by

$$\dot{x}_i(t) = -x_i(t) + b_{\hat{i}} - \sum_{j=1}^{N} l_{ij} x_j(t), \quad 1 \leqslant i \leqslant N - l_n, \tag{5.7}$$

and

$$\dot{x}_i(t) = -\sum_{j=1}^{N} l_{ij} x_j(t), \quad N - l_n + 1 \leqslant i \leqslant N. \tag{5.8}$$

The system dynamics can be written in a compact form

$$\dot{x}(t) = -\bar{L} x(t) + \bar{b}, \tag{5.9}$$

where

$$\bar{L} = \begin{bmatrix} L_{11}+I & \cdots & L_{1,n-1} & L_{1n} \\ \vdots & \ddots & \vdots & \vdots \\ L_{n-1,1} & \cdots & L_{n-1,n-1}+I & L_{n-1,n} \\ L_{n1} & \cdots & L_{n,n-1} & L_{nn} \end{bmatrix},$$

and $\bar{b} = [b_1\mathbf{1}_{l_1}^T, \ldots, b_{n-1}\mathbf{1}_{l_{n-1}}^T, \mathbf{0}_{l_n}^T]^T$.

5.2.2. Lemma. *$\bar{L}$ is invertible if and only if for any naive agent, there is a directed path from some informed agent.*

Proof: (Sufficiency) Assume that $|\bar{L}| = 0$. Then $\bar{L}x = 0$ has a non-trivial solution $x_1, \ldots, x_N$. Let r be one of the indices for which $|x_i|$, $i = 1, \ldots, N$, is maximum. Then $|x_i| < |x_r|$, for $1 \leqslant i \leqslant l_1 + \cdots + l_{n-1}$. Suppose that the contrary is true. Then consider the ith row of $\bar{L}x$. One has

$$(l_{ii}+1)|x_r| = (l_{ii}+1)|x_i| \leqslant -\sum_{j\neq i} l_{ij}|x_j| \leqslant -\sum_{j\neq i} l_{ij}|x_r|.$$

It follows that $|x_r| \leqslant 0$, which contradicts the fact that $|x_r| > 0$. We conclude that $r > l_1 + \cdots + l_{n-1}$.

For any k satisfying $|x_r| > |x_k|$, one has $l_{rk} = 0$. Otherwise, consider the rth row of $\bar{L}x$; one has

$$l_{rr}|x_r| \leqslant -\sum_{j\neq r} l_{rj}|x_j| < -\sum_{j\neq r} l_{rj}|x_r| = l_{rr}|x_r|,$$

which is a contradiction.

Let s be the number of indices j for which $|x_j| = |x_r|$. Then the rth row contains $N-s$ zeros and $l_{rk} = 0$, for $1 \leqslant k \leqslant l_1 + \cdots + l_{n-1}$. All the s corresponding rows contain $N-s$ zeros in the same places. So by the same permutations of the rows and columns, matrix $\bar{L}$ can be transformed to

$$\begin{bmatrix} U_{11} & U_{12} \\ 0 & U_{22} \end{bmatrix}, \tag{5.10}$$

where $U_{22} \in \mathbb{R}^{s\times s}$ is a square matrix and U_{11} contains

$$\begin{bmatrix} L_{11}+I & \cdots & L_{1,n-1} \\ \vdots & \ddots & \vdots \\ L_{n-1,1} & \cdots & L_{n-1,n-1}+I \end{bmatrix}$$

as a sub-matrix in the upper left corner. Thus there is no directed path from any informed agent to the naive agent in the block U_{22}.

(Necessity) If for s naive agents, there are no directed paths from any informed agent, then $\bar{L}$ can be transformed to (5.10) by the same permutations of the rows and columns such that U_{22} only contains s naive agents. U_{22} having zero row sum implies that $|\bar{L}| = 0$, which is a contradiction. □

In what follows, we assume that for any naive agent there is always a directed path from some informed agent. Similar to the system consisting of only informed agents, since $\bar{L}$ is invertible, the equilibrium x^* of system (5.9) is $x^* = \bar{L}^{-1}\bar{b}$. Let $y(t) = x(t) - x^*$; then one has $\dot{y}(t) = -\bar{L}y(t)$. It is obvious that $y(t) \to 0$ as $t \to \infty$. Thus x^* is a globally stable equilibrium of system (5.9).

In order to ensure that agents in the same cluster have the same final values, we require the following. Suppose that $-\bar{L}_{ij}$ have constant row sums r_{ij} for $i = 1, \dots, n-1,\ j = 1, \dots, n$, and that the ith row sums of $-L_{n1}, \dots, -L_{n,n-1}$ are $m_i h_1, \dots, m_i h_{n-1}$ for $1 \leqslant i \leqslant l_n$, where m_i are positive constants. We require that there is at least one $h_i \neq 0$, $1 \leqslant i \leqslant n-1$. Without loss of generality, suppose that $h_1, \dots, h_k \neq 0,\ 1 \leqslant k \leqslant n-1$, and $h_{k+1} = \cdots = h_{n-1} = 0$; it is easy to see that the ith row sums of $-L_{nn}$ are $-m_i \sum_{j=1}^{n-1} h_j$. Expanding the equation $-Q\bar{L} = I$, following a similar argument as in the proof of Lemma 5.2.1, one has

$$\left(\begin{bmatrix} r_{11} & \cdots & r_{n-1,1} & h_1 \\ r_{12} & \cdots & r_{n-1,2} & h_2 \\ \vdots & \vdots & \ddots & \vdots \\ r_{1n} & \cdots & r_{n-1,n} & -\sum_{j=1}^{n-1} h_j \end{bmatrix} \otimes I\right) \begin{bmatrix} Q_{11}\mathbf{1} \\ Q_{12}\mathbf{1} \\ \vdots \\ Q_{1n}\mathbf{m} \end{bmatrix} = \begin{bmatrix} \mathbf{1} \\ \mathbf{0} \\ \vdots \\ \mathbf{0} \end{bmatrix},$$

where $\mathbf{m} \triangleq [m_1, \dots, m_{l_n}]^T$. Let

$$M = \begin{bmatrix} h_2 r_{11} - h_1 r_{12} & \cdots & h_2 r_{n-1,1} - h_1 r_{n-1,2} \\ \vdots & \cdots & \vdots \\ h_k r_{11} - h_1 r_{1k} & \ddots & h_k r_{n-1,1} - h_1 r_{n-1,k} \\ \vdots & \ddots & \vdots \\ r_{1,n-1} & \cdots & r_{n-1,n-1} \\ -1 & \cdots & -1 \end{bmatrix},$$

then we have

$$(M \otimes I) \begin{bmatrix} Q_{11}\mathbf{1} \\ Q_{12}\mathbf{1} \\ \vdots \\ Q_{1,n-1}\mathbf{1} \end{bmatrix} = \begin{bmatrix} h_2\mathbf{1} \\ \vdots \\ h_k\mathbf{1} \\ \vdots \\ \mathbf{0} \\ \mathbf{1} \end{bmatrix}.$$

M is invertible since $-\bar{L}$ is. Then we can conclude that Q_{1j} have constant row sums for $1 \leqslant j \leqslant n-1$. In addition, the row sums s_{1j} of Q_{1j} satisfy

$$M[s_{11}, s_{12}, \dots, s_{1,n-1}]^T = [h_2, \dots, h_k, \dots, 0, 1]^T.$$

It is easy to check that $Q_{ij},\ 1 \leqslant i \leqslant n,\ 1 \leqslant j \leqslant n-1$, have constant row sums s_{ij},

$$\tilde{S} = \begin{bmatrix} s_{11} & \cdots & s_{1,n-1} \\ \vdots & \ddots & \vdots \\ s_{n-1,1} & \cdots & s_{n-1,n-1} \end{bmatrix} = \begin{bmatrix} h_2 \ \cdots \ h_k & \mathbf{0}^T & 1 \\ -h_1 I & O & \mathbf{1} \\ O & I & \mathbf{1} \end{bmatrix} M^{-T},$$

and $[s_{n1}, \dots, s_{n,n-1}] = [0, \dots, 0, 1]M^{-T}$.

So $\tilde{S}$ is invertible. For $1 \leqslant i \leqslant n-1$, $\sum_{j=1}^{n} r_{ij} = -1$, it is easy to show that $\sum_{j=1}^{n-1} s_{ij} = -1$, for $1 \leqslant i \leqslant n$. Moreover, for $1 \leqslant i \leqslant n-1$ and $1 \leqslant k \leqslant l_n$, one can derive from $-\bar{L}Q = I$ that

$$m_k h_1 s_{1i} + \cdots + m_k h_{n-1} s_{n-1,i} - m_k \sum_{j=1}^{n-1} h_j s_{ni} = 0.$$

It follows that $s_{ni} = \frac{\sum_{k=1}^{n-1} h_k s_{ki}}{\sum_{j=1}^{n-1} h_j}$.

Suppose that $\bar{x}_1, \dots, \bar{x}_n$ are the final values of the n clusters; then each cluster converges to $\bar{x}_i = -\sum_{j=1}^{n-1} s_{ij} b_j$. It follows that $[\bar{x}_1, \dots, \bar{x}_{n-1}]^T = -\tilde{S}[b_1, \dots, b_{n-1}]^T$. Since $\tilde{S}$ is invertible, using a similar argument as in the proof of Theorem 5.2.1, one can conclude that for almost all b_i with $b_i \neq b_j$ for $i \neq j$, the final values of the informed agents in different clusters are distinct from one another. In addition

$$\begin{aligned} \bar{x}_n &= -\sum_{t=1}^{n-1} s_{nt} b_t = -\sum_{t=1}^{n-1} \sum_{k=1}^{n-1} \frac{h_k s_{kt}}{\sum_{j=1}^{n-1} h_j} b_t \\ &= \sum_{k=1}^{n-1} \frac{h_k}{\sum_{j=1}^{n-1} h_j} \Big(-\sum_{t=1}^{n-1} s_{kt} b_t\Big) = \sum_{k=1}^{n-1} \frac{h_k \bar{x}_k}{\sum_{j=1}^{n-1} h_j}, \end{aligned} \tag{5.11}$$

which implies that the final values of the naive agents have to be a linear combination of the final values of the informed agents. The coefficients $h_k/\sum_{j=1}^{n-1} h_j$ are determined by the row sums of $L_{n1},\ldots,L_{n,n-1}$. Note that these final values only depend on the row sums of the sub-matrices of $\bar{L}$, but not on the number of agents and the proportion of the informed agents in the system. Hence, we have proved the following theorem.

5.2.3. Theorem. *For system (5.9), if for any naive agent there is a directed path from some informed agent, $-L_{ij}$ have constant row sums r_{ij} for $i=1,\ldots,n-1,\ j=1,\ldots,n$, and the ith row sums of $-L_{n1},\ldots,-L_{n,n-1}$ are $m_ih_1,\ldots,m_ih_{n-1}$ for some $m_i>0,\ 1\leqslant i\leqslant l_n$, then for any initial condition, the final values of the clusters of the informed agents are distinct from one another for almost all (in the sense of Lebesgue measure) b_i for $1\leqslant i\leqslant n-1$ with $b_i\neq b_j$ for $i\neq j$, and the final values of the naive agents converge to a linear combination of the asymptotic values of the informed agents as defined in (5.11).*

5.2.1. Remark. *In [60], more general agent dynamics are considered. Consequently besides the condition of constant row sums stipulated in Theorem 5.2.3, additional conditions have to be imposed to guarantee clustering. Since more restricted agent dynamics are considered here, the agents' final values can be predicted whereas it is difficult to do so for the model considered in [60].*

In this section, we have considered the clustering behavior when the agents have different linear dynamics. In the next section, we consider more challenging scenarios, in which agents are governed by the same self-dynamics.

5.3 Clustering with identical self-dynamics

Now we consider the case when all the agents have the same self-dynamics:

$$\dot{x}_i(t)=f(t,x_i(t))-c\sum_{j=1}^{N} l_{ij}\Gamma x_j(t),\quad 1\leqslant i\leqslant N, \tag{5.12}$$

where the notation is the same as in (5.1), and f is a continuous map that is globally Lipschitzian in x_i with Lipschitz constant K and $l_{ij}\leqslant 0$ for $i\neq j$. There are existing results discussing when clustering might appear in (5.12) [60, 78]. We first compare these results.

Let $\mathcal{X}$ denote the manifold $\{x=[x_1^T(t),\ldots,x_N^T(t)]^T:\ x_1(t)=\cdots=x_{l_1}(t),\ x_{l_1+1}(t)=\cdots=x_{l_1+l_2}(t),\ldots,\ x_{N-l_n+1}(t)=\cdots=x_N(t)\}$ corresponding to the n-cluster synchronization. The following result is from [60].

5.3.1. THEOREM. *[60] The manifold $\mathcal{X}$ is invariant if and only if the block matrices L_{ij} achieved by partitioning L into submatrices corresponding to the clusters have constant row sums.*

A sufficient condition for the same n-cluster synchronization manifold to be invariant is given in [78]; it can be stated as follows.

5.3.2. THEOREM. *[78] The manifold $\mathcal{X}$ is invariant if there is a solution X to the linear equations*

$$(I_N - \Pi)L = X(I_N - \Pi), \tag{5.13}$$

where Π is a permutation matrix such that $\mathcal{X} = ker(I_{mN} - \Pi \otimes I_m)$.

We now prove that the conditions given in Theorem 5.3.1 and Theorem 5.3.2 are in fact equivalent.

5.3.1. PROPOSITION. *The block matrices L_{ij} of L have constant row sums if and only if there exists a solution X to the linear equations (5.13), where Π is a permutation matrix satisfying $\mathcal{X} = ker(I_{mN} - \Pi \otimes I_m)$.*

Proof: (Necessity) Since $\mathcal{X} = ker(I_{mN} - \Pi \otimes I_m)$, $\Pi = diag\{\Pi_1, \ldots, \Pi_n\}$, where Π_i are permutation matrices with the same dimensions of L_{ii}. From (5.13), we have

$$(I - \Pi_i)L_{ij} = X_{ij}(I - \Pi_j), \qquad 1 \leqslant i, j \leqslant n. \tag{5.14}$$

Since L_{ij} have constant row sums, the row sums of $(I - \Pi_i)L_{ij}$ are zero. Suppose that X_{ij} is a $u \times v$ matrix. Let $L_{ij}^T(I - \Pi_i)^T = [\beta_1, \beta_2, \ldots, \beta_u]$ and $X_{ij}^T = [\alpha_1, \ldots, \alpha_u]$, where α_i and β_i, $1 \leqslant i \leqslant u$, are column vectors. Then (5.14) is equivalent to

$$(I - \Pi_j)^T \alpha_k = \beta_k, \qquad 1 \leqslant k \leqslant u. \tag{5.15}$$

Since $\text{rank}(I - \Pi_j)^T = \text{rank}([(I - \Pi_j)^T \ \beta_k]) = v - 1$, there exist solutions to (5.15). Then there exists a solution X to (5.13).

(Sufficiency) Without loss of generality, suppose that the permutation matrix Π can be written as $\Pi = diag\{\Pi_1, \ldots, \Pi_q, \underbrace{1, \ldots, 1}_{n-q}\}$, where Π_k, $1 \leqslant k \leqslant q$, are permutation matrices with the diagonal elements being zero. Then we can partition the matrix L into $n \times n$ blocks with the dimensions of L_{kk}, $q + 1 \leqslant k \leqslant n$, all being one. Thus we only need to prove that L_{ij}, $1 \leqslant i, j \leqslant q$, have constant row sums. Let $L_{ij} = [\theta_1, \ldots, \theta_u]^T$, where θ_i are column vectors. From (5.14), it follows that

$$(I - \Pi_i)L_{ij} = [\theta_1 - \theta_{i_1}, \ldots, \theta_u - \theta_{i_u}]^T = X_{ij}(I - \Pi_j),$$

where $\{i_1, \ldots, i_u\}$ is a permutation of $\{1, \ldots, u\}$ determined by Π_i. The row sums of $X_{ij}(I-\Pi_j)$ are zero because of the zero row sums of $I-\Pi_j$. In addition, the diagonal entries of Π_i are zero, so the row sums of θ_i^T, $1 \leqslant i \leqslant u$, are the same; namely L_{ij} have constant row sums. □

We have just compared different conditions on when $\mathcal{X}$ is invariant. To further guarantee clustering to take place, we now introduce coupling delay into the system. There are also other mechanisms that give rise to clustering behavior of the system, such as the pinning control strategy, the interested reader is referred to [104].

5.3.1 Delay-induced cluster synchronization

In view of Theorem 5.3.1, in this subsection we assume that $-L_{ij}$ have constant row sums r_{ij}, $1 \leqslant i, j \leqslant n$. We introduce a coupling delay to (5.12) as follows [59, 70]:

$$\begin{aligned}\dot{x}_i(t) &= f(t, x_i(t)) - c \sum_{j=1, j\neq i}^{N} l_{ij}\Gamma(x_j(t-\tau) - x_i(t)) \\ &= f(t, x_i(t)) - c \sum_{j=1}^{N} l_{ij}\Gamma x_j(t-\tau) + c d_i^{in}\Gamma(x_i(t-\tau) - x_i(t)),\end{aligned} \tag{5.16}$$

where the notation is the same as in (5.12), and in addition $d_i^{in} = l_{ii} = \sum_{j=1, j\neq i}^{N} a_{ij}$ is the in-degree of the ith agent, and $\tau > 0$ denotes the time delay. The initial condition for (5.16) is given by $x_i(\theta) = \phi_i(\theta)$, for $1 \leqslant i \leqslant N$, $\theta \in [-\tau, 0]$, where $\phi_i(\theta) \in C([-\tau, 0], \mathbb{R}^m)$. Since f is a continuous map that is globally Lipschitzian in x_i, and the couplings among agents are linear, system (5.16) has a unique solution which exists for all $t \geqslant 0$ [33].

When the N coupled agents achieve complete synchronization, i.e. $x_1(t) = x_2(t) = \cdots = x_N(t) = s(t)$, we have the following synchronized state equation:

$$\dot{s}(t) = f(t, s(t)) + c d_i^{in}\Gamma(s(t-\tau) - s(t)), i = 1, \ldots, N. \tag{5.17}$$

When $s(t-\tau) \neq s(t)$, a necessary condition for the synchronization manifold to be invariant is that $d_1^{in} = d_2^{in} = \cdots = d_N^{in}$. When the N coupled agents achieve n-cluster synchronization, i.e., $x_i(t) = x_j(t) = s_{\hat{i}}(t)$ for $\hat{i} = \hat{j}$, and $s_i(t) \neq s_j(t)$ for $\hat{i} \neq \hat{j}$, we have

$$\dot{s}_{\hat{i}}(t) = f(t, s_{\hat{i}}(t)) - c \sum_{j\neq i, j\in C_{\hat{i}}} l_{ij}\Gamma(s_{\hat{i}}(t-\tau) - s_{\hat{i}}(t)) + c \sum_{k=1, k\neq \hat{i}}^{n} r_{\hat{i}k}\Gamma(s_k(t-\tau) - s_{\hat{i}}(t)),$$

Then a necessary condition for the cluster synchronization manifold to be invariant is that $d_i^{in} = d_j^{in}$ for $\hat{i} = \hat{j}$ and $d_i^{in} \neq d_j^{in}$ for $\hat{i} \neq \hat{j}$.

Let $D^{in} = diag\{d_1^{in}, \ldots, d_N^{in}\}$. Assume that the graph $\mathbb{G}$ representing the interaction topology among the agents is strongly connected; then L is irreducible. Hence, zero is a simple eigenvalue of L associated with a positive left eigenvector $\xi = [\xi_1, \xi_2, \ldots, \xi_N]^T$. Define $E = diag\{\xi_1, \ldots, \xi_N\}$.

Now consider the ith agent. Define the average of the $\hat{i}$th cluster to be

$$\bar{x}_{\hat{i}}(t) = \frac{1}{\sum_{i \in C_{\hat{i}}} \xi_i} \sum_{i \in C_{\hat{i}}} \xi_i x_i(t),$$

and the difference between agent i's state and this average to be $e_i(t) = x_i(t) - \bar{x}_{\hat{i}}(t)$. Then

$$\begin{aligned}
\dot{e}_i(t) &= \dot{x}_i(t) - \dot{\bar{x}}_{\hat{i}}(t) \\
&= f(x_i(t)) - c \sum_{j=1}^{N} l_{ij} \Gamma x_j(t-\tau) \\
&\quad + c d_i^{in} \Gamma (x_i(t-\tau) - x_i(t)) - \dot{\bar{x}}_{\hat{i}}, \ i = 1, \ldots, N.
\end{aligned} \tag{5.18}$$

Let $e_i(t) = [e_{i1}(t), e_{i2}(t), \ldots, e_{im}(t)]^T \in \mathbb{R}^m$, $e(t) = [e_1^T(t), \ldots, e_N^T(t)]^T$, $\tilde{e}_i(t) = [e_{1i}(t), e_{2i}(t), \ldots, e_{Ni}(t)]^T \in \mathbb{R}^N$ and $\tilde{e}(t) = [\tilde{e}_1^T(t), \ldots, \tilde{e}_m^T(t)]^T$. Then one can check that

$$\sum_{i \in C_{\hat{i}}} \xi_i e_i = \sum_{i \in C_{\hat{i}}} \xi_i x_i - \sum_{i \in C_{\hat{i}}} \xi_i \left(\frac{1}{\sum_{i \in C_{\hat{i}}} \xi_i} \right) \sum_{i \in C_{\hat{i}}} \xi_i x_i = 0.$$

Hence,

$$\sum_{i \in C_{\hat{i}}} \xi_i e_i^T \dot{\bar{x}}_{\hat{i}}(t) = 0, \ \sum_{i \in C_{\hat{i}}} \xi_i e_i^T f(t, \bar{x}_{\hat{i}}(t)) = 0,$$

$$\sum_{i \in C_{\hat{i}}} \xi_i e_i^T \left(c d_i^{in} \Gamma (\bar{x}_{\hat{i}}(t-\tau) - \bar{x}_{\hat{i}}(t)) \right) = 0, \ \sum_{i \in C_{\hat{i}}} \xi_i e_i^T \left(\sum_{k=1}^{n} \sum_{j \in C_k} l_{ij} \Gamma \bar{x}_k(t) \right) = 0.$$

Since $f(t, x)$ satisfies the Lipschitz condition (5.2), there must exist a diagonal matrix $\Delta = diag\{\delta_1, \ldots, \delta_m\}$ such that

$$(x-y)^T (f(t,x) - f(t,y) - \Delta(x-y)) \leqslant -\alpha (x-y)^T (x-y) \tag{5.19}$$

holds for some $\alpha > 0$, all $x, y \in \mathbb{R}^m$ and all $t \geqslant 0$. A simple choice of Δ is $(K+\alpha)I$, while for a specific $f(t,x)$ of interest, less conservative Δ can be found. Now we present the main result in this subsection.

5.3.3. THEOREM. *Suppose that $-L_{ij}$ have constant row sums r_{ij}, for $i, j = 1, \ldots, n$, that the in-degree d_i^{in} of each agent satisfies $d_i^{in} = d_j^{in}$ for $\hat{i} = \hat{j}$ and $d_i^{in} \neq d_j^{in}$ for*

$\hat{i} \neq \hat{j}$, and that Δ is a diagonal matrix satisfying (5.19). If there exist positive definite matrices $Q_j \succ 0$ such that the linear matrix inequalities

$$\begin{bmatrix} 2\delta_j E - 2c\gamma_j ED^{in} + Q_j & c\gamma_j E(D^{in} - L) \\ c\gamma_j (D^{in} - L^T)E & -Q_j \end{bmatrix} \prec 0 \tag{5.20}$$

hold for all $j = 1, \ldots, m$, then $e_i(t) \to 0$ as $t \to \infty$ for $i = 1, \ldots, N$.

Proof: Since the matrix inequalities (5.20) are valid, there exists a positive constant ϵ such that $-2\alpha + \epsilon < 0$ and

$$\Lambda_j = \begin{bmatrix} 2\delta_j E - 2c\gamma_j ED^{in} + Q_j e^{\epsilon\tau} & c\gamma_j E(D^{in} - L) \\ c\gamma_j (D^{in} - L^T)E & -Q_j \end{bmatrix} \prec 0$$

hold for all $j = 1, \ldots, m$. Let

$$\begin{aligned} V_1 &= \sum_{\hat{i}=1}^{n} W_{\hat{i}}(t) = \sum_{\hat{i}=1}^{n} \sum_{i \in C_{\hat{i}}} \xi_i e_i^T(t) e_i(t) e^{\epsilon t}, \\ V_2 &= \sum_{j=1}^{m} \int_{t-\tau}^{t} \tilde{e}_j^T(s) Q_j \tilde{e}_j(s) e^{\epsilon(s+\tau)} ds. \end{aligned}$$

Consider the candidate Lyapunov function $V(t) = V_1(t) + V_2(t)$. Then, for $W_{\hat{i}}(t) = \sum_{i \in C_{\hat{i}}} \xi_i e_i^T(t) e_i(t) e^{\epsilon t}$, its derivative along the solutions to (5.18) is

$$\begin{aligned} \dot{W}_{\hat{i}} &= 2e^{\epsilon t} \sum_{i \in C_{\hat{i}}} \xi_i e_i^T(t) \Big(f(t, x_i(t)) - f(t, \bar{x}_{\hat{i}}(t)) + f(t, \bar{x}_{\hat{i}}(t)) - \Delta e_i(t) \\ &\quad + \Delta e_i(t) - c \sum_{j=1}^{N} l_{ij} \Gamma (x_j(t-\tau) - \bar{x}_{\hat{j}}(t-\tau)) - c \sum_{k=1}^{n} \sum_{j \in C_k} l_{ij} \Gamma \bar{x}_k(t-\tau) \\ &\quad + c d_i^{in} \Gamma (x_i(t-\tau) - x_i(t)) - c d_i^{in} \Gamma (\bar{x}_{\hat{i}}(t-\tau) - \bar{x}_{\hat{i}}(t)) \\ &\quad + c d_i^{in} \Gamma (\bar{x}_{\hat{i}}(t-\tau) - \bar{x}_{\hat{i}}(t)) - \dot{\bar{x}}_{\hat{i}}(t) \Big) + \epsilon e^{\epsilon t} \sum_{i \in C_{\hat{i}}} \xi_i e_i^T(t) e_i(t) \\ &= 2e^{\epsilon t} \sum_{i \in C_{\hat{i}}} \xi_i e_i^T(t) \Big(f(t, x_i(t)) - f(t, \bar{x}_{\hat{i}}(t)) - \Delta e_i(t) + \Delta e_i(t) \\ &\quad - c \sum_{j=1}^{N} l_{ij} \Gamma e_j(t-\tau) + c d_i^{in} \Gamma (e_i(t-\tau) - e_i(t)) \Big) + \epsilon e^{\epsilon t} \sum_{i \in C_{\hat{i}}} \xi_i e_i^T(t) e_i(t) \\ &\leqslant (-2\alpha + \epsilon) e^{\epsilon t} \sum_{i \in C_{\hat{i}}} \xi_i e_i^T(t) e_i(t) \\ &\quad + 2e^{\epsilon t} \sum_{i \in C_{\hat{i}}} \xi_i e_i^T(t) \Big(\Delta e_i(t) - c \sum_{j=1}^{N} l_{ij} \Gamma e_j(t-\tau) + c d_i^{in} \Gamma (e_i(t-\tau) - e_i(t)) \Big). \end{aligned}$$

Then, it follows that

$$\begin{aligned}
& \dot{V}(t) \\
\leqslant\; & (-2\alpha+\epsilon)e^{\epsilon t}\sum_{\hat{i}=1}^{n}\sum_{i\in C_{\hat{i}}}\xi_i e_i^T(t)e_i(t) \\
& +2e^{\epsilon t}\sum_{\hat{i}=1}^{n}\sum_{i\in C_{\hat{i}}}\xi_i e_i^T(t)\Big(\Delta e_i(t)-c\sum_{j=1}^{N}l_{ij}\Gamma e_j(t-\tau)+cd_i^{in}\Gamma(e_i(t-\tau)-e_i(t))\Big) \\
& +e^{\epsilon(t+\tau)}\sum_{j=1}^{m}\tilde{e}_j^T(t)Q_j\tilde{e}_j(t)-e^{\epsilon t}\sum_{j=1}^{m}\tilde{e}_j^T(t-\tau)Q_j\tilde{e}_j(t-\tau) \\
\leqslant\; & (-2\alpha+\epsilon)e^{\epsilon t}\sum_{\hat{i}=1}^{n}\sum_{i\in C_{\hat{i}}}\xi_i e_i^T(t)e_i(t)-e^{\epsilon t}\sum_{j=1}^{m}\tilde{e}_j^T(t-\tau)Q_j\tilde{e}_j(t-\tau) \\
& +e^{\epsilon t}\sum_{j=1}^{m}\tilde{e}_j^T(t)\Big((2\delta_j E-2c\gamma_j ED^{in}+Q_j e^{\epsilon\tau})\tilde{e}_j(t)+2c\gamma_j E(D^{in}-L)\tilde{e}_j(t-\tau)\Big) \\
=\; & (-2\alpha+\epsilon)e^{\epsilon t}\sum_{\hat{i}=1}^{n}\sum_{i\in C_{\hat{i}}}\xi_i e_i^T(t)e_i(t)+e^{\epsilon t}\sum_{j=1}^{m}[\tilde{e}_j^T(t),\tilde{e}_j^T(t-\tau)]\Lambda_j\begin{bmatrix}\tilde{e}_j(t)\\ \tilde{e}_j(t-\tau)\end{bmatrix} \\
\leqslant\; & 0.
\end{aligned}$$

Therefore, $V(t)\leqslant V(0)$ which implies that $V_1(t)$ is bounded. In view of the definition of V_1, this further implies that $||e(t)||_2$ is bounded from above by an exponentially decaying signal that converges to zero. This completes the proof. □

Theorem 5.3.3 has shown that the differences among the states of the agents in the same cluster will converge to zero as time goes to infinity. However, it is in general difficult to prove that the differences between clusters do not converge to zero. Next we prove 2-cluster synchronization when f is periodic. Consider

$$f(t,x_i(t)) = Bx_i(t)+h(x_i(t))+\beta(t), \tag{5.21}$$

where $B=diag\{b_1,\dots,b_m\}$ with negative constants $b_i<0$, $\beta:[0,\infty)\to\mathbb{R}^m$ is a continuous, periodic function with period $\omega>0$, i.e., $\beta(t+\omega)=\beta(t)$, and $h:\mathbb{R}^m\to\mathbb{R}^m$ is a bounded function which satisfies $||h(\xi_1)-h(\xi_2)||_2\leqslant H||\xi_1-\xi_2||_2$. We first present the following result.

5.3.1. LEMMA. *If there exist positive definite matrices P_j such that the linear matrix inequalities*

$$\begin{bmatrix}2(b_j+H)I-2c\gamma_j D^{in}+P_j & c\gamma_j(D^{in}-L)\\ c\gamma_j(D^{in}-L^T) & -P_j\end{bmatrix}\prec 0 \tag{5.22}$$

hold for all $j = 1, \ldots, m$, then the coupled system (5.16) with f in the form of (5.21) has exactly one periodic solution with period ω to which all the other solutions converge exponentially fast as $t \to \infty$.

Proof: Let $\mathcal{C} = C([-\tau, 0], \mathbb{R}^m)$. For any $\phi_i \in \mathcal{C}$, we define $||\phi_i||_\tau = \sup_{-\tau \leqslant \theta \leqslant 0} ||\phi_i(\theta)||_2$. For any $\phi = [\phi_1^T, \ldots, \phi_N^T]^T$, where $\phi_i \in \mathcal{C}$, $1 \leqslant i \leqslant N$, we denote the solution of (5.16) through $(0, \phi)$ as $x(t, \phi) = [x_1^T(t, \phi), \ldots, x_N^T(t, \phi)]^T$, and define $x_t(\phi) = x(t+\theta, \phi)$, $\theta \in [-\tau, 0]$, $t \geqslant 0$; then $x_t(\phi) \in \mathcal{C}$ for all $t \geqslant 0$.

Now consider two solutions $x(t, \phi)$ and $x(t, \varphi)$ of (5.16). Define $w_i(t) = x_i(t, \phi) - x_i(t, \varphi)$, $w(t) = [w_1^T(t), \ldots, w_N^T(t)]^T$, $\tilde{w}_i(t) = [w_{1i}(t), \ldots, w_{Ni}(t)]^T$, and $\tilde{w}(t) = [\tilde{w}_1^T(t), \ldots, \tilde{w}_m^T(t)]^T$. It follows from (5.16) and (5.21) that

$$\begin{aligned} \dot{w}_i(t) &= Bw_i(t) + h(x_i(t, \phi)) - h(x_i(t, \varphi)) \\ &\quad -c \sum_{j=1}^{N} l_{ij} \Gamma w_j(t - \tau) + c d_i^{in} \Gamma (w_i(t - \tau) - w_i(t)). \end{aligned}$$

Since the matrix inequalities (5.22) are valid, there exists a positive constant ϵ such that

$$\Omega_j = \begin{bmatrix} 2(b_j + H)I + \epsilon I - 2c\gamma_j D^{in} + P_j e^{\epsilon\tau} & c\gamma_j (D^{in} - L) \\ c\gamma_j (D^{in} - L^T) & -P_j \end{bmatrix}$$

are negative definite for all $j = 1, \ldots, m$. Consider the candidate Lyapunov function

$$V(t) = \sum_{i=1}^{N} w_i^T(t) w_i(t) e^{\epsilon t} + \sum_{j=1}^{m} \int_{t-\tau}^{t} \tilde{w}_j^T(s) P_j \tilde{w}_j(s) e^{\epsilon(s+\tau)} ds.$$

By similar calculations to the proof of Theorem 5.3.3, we obtain

$$\dot{V}(t) \leqslant e^{\epsilon t} \sum_{j=1}^{m} [\tilde{w}_j^T(t), \tilde{w}_j^T(t-\tau)] \Omega_j \begin{bmatrix} \tilde{w}_j(t) \\ \tilde{w}_j(t - \tau) \end{bmatrix} \leqslant 0.$$

Therefore, $V(t) \leqslant V(0)$, from which it follows that

$$||x(t, \phi) - x(t, \varphi)||_2 \leqslant M e^{-\frac{\epsilon}{2} t} ||\phi - \varphi||_\tau, \ t \geqslant 0,$$

where $M \geqslant 1$ is a constant. Then, it is easy to see that

$$||x_t(\phi) - x_t(\varphi)||_\tau \leqslant M e^{-\frac{\epsilon}{2}(t-\tau)} ||\phi - \varphi||_\tau. \tag{5.23}$$

Comparing (5.23) and equation (5) in [14], it is easy to see that using similar arguments to that in [14], one can conclude that system (5.16) has exactly one periodic solution with period ω and all the other solutions converge exponentially to it as $t \to \infty$. □

With Lemma 5.3.1, we now prove 2-cluster synchronization.

5.3.4. Theorem. *Suppose that $-L_{ij}$ have constant row sums r_{ij}, for $i, j = 1, \ldots, n$, $\tau \neq k\omega$, for $k \geqslant 0$, and $d_i^{in} = d_j^{in}$ for $\hat{\imath} = \hat{\jmath}$ and $d_i^{in} \neq d_j^{in}$ for $\hat{\imath} \neq \hat{\jmath}$. If there exist positive definite matrices P_j and Q_j such that (5.22) and (5.20) hold with $\delta_j = b_j + H$ for $j = 1, \ldots, m$, then for any initial condition, the coupled system (5.16) with f in the form of (5.21) realizes 2-cluster synchronization.*

Proof: In view of Theorem 5.3.3, we only need to show that complete synchronization cannot be achieved. Suppose that the contrary is true. Then (5.17) holds for all $i = 1, \ldots, N$. It follows from Lemma 5.3.1 that $s(t)$ is a periodic function with period ω. Since $\tau \neq k\omega$ for $k \geqslant 0$, it follows that $s(t-\tau)$ cannot be equal to $s(t)$ for all t. Thus we have $d_1^{in} = d_N^{in}$, which contradicts the fact that $d_1^{in} \neq d_N^{in}$ since agents 1 and N do not belong to the same cluster. □

However, we are unable to prove n-cluster synchronization for $n \geqslant 3$ using the idea of Theorem 5.3.4 due to difficulties in showing that the difference between the states of any two different clusters will not converge to 0. To prove this, we need to show that k-cluster synchronization cannot happen for all $k = 1, \ldots, n-1$, which becomes involved when n is large. In the 2-cluster synchronization case, this reduces to show that complete synchronization cannot be achieved in the system, which simplifies the analysis. We show through simulations in Section 5.4 that n-cluster synchronization can be achieved if (5.20) and (5.22) are satisfied for $n \geqslant 3$.

In the next subsection, we discuss a different approach to realize cluster synchronization when the agents' self-dynamics are identical.

5.3.2 Clustering with negative couplings

In this subsection, we study how clustering may appear as a pure effect of structured diffusive couplings. We assume that the agents' dynamics are completely determined by their couplings:

$$\dot{x}_i(t) = -\sum_{j=1}^{N} l_{ij} x_j(t), \tag{5.24}$$

or in a compact form

$$\dot{x}(t) = -Lx(t), \tag{5.25}$$

Comparing to (5.1), we have taken Γ to be an identity matrix, $c = 1$, and $m = 1$. The results derived below can be easily extended to the general case when $c > 0$ and $m \geqslant 1$. From Theorem 2.4.1 we know that, if the weights of the edges in $\mathbb{G}$ are all positive, i.e., $a_{ij} = -l_{ij} \geqslant 0$, $i \neq j$, and $\mathbb{G}$ contains a directed spanning tree, then the system achieves consensus. In this subsection, we assume that there might be negative couplings in the graph $\mathbb{G}$, and as a result $\mathbb{G}$ is a signed graph. The adjacency matrix A becomes a signed adjacency matrix, but the Laplacian matrix L is still defined as in (2.1) here. The approach that system (5.24) models negative couplings is different from that in (4.19), which consequently gives rise to different dynamical behavior. Since the signed Laplacian matrix considered in (4.20) is a diagonally dominant matrix, its eigenvalues all have nonnegative real parts which guarantees the convergence of system (4.20). Furthermore, the system states polarize if zero is an eigenvalue of the signed Laplacian matrix and converge to zero otherwise. However, the convergence of system (5.25) is not guaranteed since the Laplacian matrix in (5.25) may have eigenvalues with negative real parts. Zero is always an eigenvalue. In addition, the geometric multiplicity of the eigenvalue zero and thus the dimension of the null space of the Laplacian matrix, can be larger than one. This allows the possibility of the occurrence of clustering in the system as we will see later.

Let $\eta_1 = [\mathbf{1}_{l_1}^T, \mathbf{0}_{N-l_1}^T]^T$, $\eta_2 = [\mathbf{0}_{l_1}^T, \mathbf{1}_{l_2}^T, \mathbf{0}_{N-l_1-l_2}^T]^T$, $\ldots, \eta_n = [\mathbf{0}_{N-l_n}^T, \mathbf{1}_{l_n}^T]^T$, and let $\alpha_1, \ldots, \alpha_n$ be n independent vectors satisfying $\eta_i^T \alpha_j = 1$, if $i = j$ and $\eta_i^T \alpha_j = 0$, if $i \neq j$. Since the solution to (5.25) is $x(t) = e^{-Lt}x(0)$, it is obvious that if

$$\lim_{t \to \infty} e^{-Lt} = \sum_{i=1}^{n} \eta_i \alpha_i^T, \tag{5.26}$$

then n-cluster synchronization might be achieved. We provide the following necessary and sufficient condition under which (5.26) holds.

5.3.2. Lemma. *Equation (5.26) holds if and only if*

$$L\eta_i = 0, \ \alpha_i^T L = 0, \qquad i = 1, \ldots, n, \tag{5.27}$$

where $-L$ has exactly n zero eigenvalues and all the other eigenvalues have negative real parts.

Proof: We give the proof for the case when $n = 2$. The proof for the general case $n \geqslant 2$ can be proved following similar steps.

(Sufficiency) This has been proved as Lemma 6 in [112].

(Necessity) Let $J = diag\{J_1, \ldots, J_s\}$ be the Jordan form of $-L$, i.e., there exists a nonsingular matrix P such that $-L = PJP^{-1}$. Then

$$\lim_{t \to \infty} e^{-Lt} = P \lim_{t \to \infty} diag\{e^{J_1 t}, \ldots, e^{J_s t}\} P^{-1}.$$

$\lim_{t \to \infty} e^{-Lt}$ exists if and only if J_i are zero matrices or the eigenvalues of J_i have negative real parts. Let $u_1, \ldots, u_N$ be the columns of P and let $v_1^T, \ldots, v_N^T$ be the rows of P^{-1}. Then the fact that (5.26) holds implies that J has the form $J = diag\{O_k, Z\}$, where the eigenvalues of Z have negative real parts. We have

$$\lim_{t\to\infty} e^{-Lt} = P \left[\begin{array}{cc} I_k & 0 \\ 0 & 0 \end{array} \right] P^{-1} = \sum_{i=1}^{k} u_i v_i^T.$$

Since $\text{rank}(u_i v_i^T) = 1$ and $\text{rank}(\sum_{i=1}^N u_i v_i^T = I) = N$, $\sum_{i=1}^k u_i v_i^T$ must have rank k. Combined with (5.26), one has $k = 2$ and $u_1 v_1^T + u_2 v_2^T = \eta_1 \alpha_1^T + \eta_2 \alpha_2^T$, which implies that $-L$ has exactly two zero eigenvalues and all the other eigenvalues have negative real parts. In addition, one has

$$u_{11} v_1^T + u_{21} v_2^T = \alpha_1^T, \qquad \cdots, \qquad u_{1l_1} v_1^T + u_{2l_1} v_2^T = \alpha_1^T,$$

which implies that $(u_{1i} - u_{1j}) v_1^T + (u_{2i} - u_{2j}) v_2^T = 0$. Then $u_{1i} = u_{1j}$ and $u_{2i} = u_{2j}$ for $1 \leqslant i, j \leqslant l_1$. Using similar arguments we have

$$u_{11} = \cdots = u_{1l_1}, \; u_{1l_1+1} = \cdots = u_{1N},$$
$$u_{21} = \cdots = u_{2l_1}, \; u_{2l_1+1} = \cdots = u_{2N}.$$

If $u_{11} = 0$, then $[\mathbf{0}_{l_1}^T, \mathbf{1}_{l_2}^T]^T$ is a right eigenvector associated with 0, and so is $[\mathbf{1}_{l_1}^T, \mathbf{0}_{l_2}^T]^T$. If $u_{11} \neq 0$, $[\mathbf{0}_{l_1}^T, \frac{u_{2N} u_{11} - u_{1N} u_{21}}{u_{11}} \mathbf{1}_{l_2}^T]^T$ is a right eigenvector associated with 0. So $[\mathbf{0}_{l_1}^T, \mathbf{1}_{l_2}^T]^T$ and $[\mathbf{1}_{l_1}^T, \mathbf{0}_{l_2}^T]^T$ are right eigenvectors associated with 0.

Without loss of generality, choose $u_1 = \eta_1 = [\mathbf{1}_{l_1}^T, \mathbf{0}_{l_2}^T]^T$ and $u_2 = \eta_2 = [\mathbf{0}_{l_1}^T, \mathbf{1}_{l_2}^T]^T$; then $\eta_1 (v_1 - \alpha_1)^T + \eta_2 (v_2 - \alpha_2)^T = 0$, which implies that $v_1 = \alpha_1$ and $v_2 = \alpha_2$. Hence, one has $\alpha_1^T L = \alpha_2^T L = 0$. □

From Lemma 5.3.2, it is clear that in order to realize n-cluster synchronization, L_{ij} have to have zero row sums. In the following discussion, assume that L satisfies the condition that the row sums of L_{ij}, $1 \leqslant i, j \leqslant n$, are zero, then L has zero as an eigenvalue whose geometric multiplicity is at least n. Let $\eta_1 = [\mathbf{1}_{l_1}^T, \mathbf{0}_{N-l_1}^T]^T, \ldots, \eta_n = [\mathbf{0}_{N-l_n}^T, \mathbf{1}_{l_n}^T]^T$, be n right eigenvectors associated with 0, and let $\alpha_1, \ldots, \alpha_n$ be the corresponding left eigenvectors satisfying $\eta_i^T \alpha_j = 1$, if $i = j$, and $\eta_i^T \alpha_j = 0$, if $i \neq j$. The following result is a slightly modified version of the main result of [112].

5.3.5. Theorem. *Suppose that the initial values of system (5.25) satisfy that $\alpha_i^T x(0)$ with $1 \leqslant i \leqslant n$ are not equal to each other; then n-cluster synchronization can be achieved if and only if $-L$ has exactly n zero eigenvalues and all the other eigenvalues have negative real parts.*

The conditions stipulated in Theorem 5.3.5 for achieving n-cluster synchronization is an algebraic condition, which is difficult to check in application. Now we develop algorithms to construct appropriate coupling topologies which satisfy the conditions in Theorem 5.3.5.

5.3.3. LEMMA. *[44] Let A and B be $N \times N$ Hermitian matrices and let the eigenvalues $\lambda_i(A)$, $\lambda_i(B)$, and $\lambda_i(A+B)$ be arranged in decreasing order as $\lambda_N(\cdot) \leqslant \lambda_{N-1}(\cdot) \leqslant \cdots \leqslant \lambda_1(\cdot)$. For each $k = 1, 2, \cdots, N$, we have*

$$\lambda_k(A) + \lambda_N(B) \leqslant \lambda_k(A+B) \leqslant \lambda_k(A) + \lambda_1(B).$$

Intuitively, if the inner couplings within the clusters are strong enough, system (5.25) can achieve cluster synchronization. This is verified by the following results.

5.3.2. PROPOSITION. *Let*

$$L = diag\{c_1 L_{11}, \ldots, c_n L_{nn}\} + \begin{bmatrix} 0 & L_{12} & \cdots & L_{1n} \\ L_{21} & 0 & \cdots & L_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ L_{n1} & L_{n2} & \cdots & 0 \end{bmatrix}$$

be a symmetric Laplacian matrix, $L_1 = diag\{c_1 L_{11}, \ldots, c_n L_{nn}\}$, and $L_2 = L - L_1$. Suppose that L_{ij} have zero row sums, matrices L_{ii} are irreducible and the off-diagonal elements of $-L_{ii}$ are nonnegative. If $c_i > \frac{\rho(L_2)}{-\max_{1 \leqslant i \leqslant n} \lambda_2(-L_{ii})}$, then $-L$ has exactly n zero eigenvalues and all the other eigenvalues are negative.

Proof: Since L_{ij} have zero row sums, L has at least n zero eigenvalues. Using Lemma 5.3.3, one has

$$\lambda_N(-L_2) \leqslant \lambda_i(-L) - \lambda_i(-L_1) \leqslant \lambda_1(-L_2),$$

which leads to $|\lambda_i(-L) - \lambda_i(-L_1)| \leqslant \rho(-L_2) = \rho(L_2)$. It follows from $c_i > \rho(L_2)/ (-\max_{1 \leqslant i \leqslant n} \lambda_2(-L_{ii}))$ that $\max_{1 \leqslant i \leqslant n} c_i \lambda_2(-L_{ii}) + \rho(L_2) < 0$. Since L_{ii} are irreducible and the off-diagonal elements of $-L_{ii}$ are nonnegative, it follows that $\lambda_1(-L_1) = \cdots = \lambda_n(-L_1) = 0$, and $\lambda_{n+1}(-L_1) = \max_{1 \leqslant i \leqslant n} c_i \lambda_2(-L_{ii})$. Thus one concludes that $\lambda_{n+1}(-L) \leqslant \max\limits_{1 \leqslant i \leqslant n} c_i \lambda_2(-L_{ii}) + \rho(L_2) < 0$. □

5.3.3. PROPOSITION. *Suppose that the graphs $\mathbb{G}_1, \ldots, \mathbb{G}_n$ are balanced and strongly connected and the weights of the edges in these graphs are positive. Assume that $L_1, \ldots, L_n$ are the corresponding Laplacian matrices. For any positive definite matrix S with proper dimension, zero is an eigenvalue of $-Sdiag\{L_1, \ldots, L_n\}$ of algebraic and geometric multiplicity n, and all the other eigenvalues of $-Sdiag\{L_1, \ldots, L_n\}$ have negative real parts.*

Proposition 5.3.3 can be proved using a similar argument to that in the proof of Theorem 4.5 in [55].

Proposition 5.3.3 provides a way to construct a graph satisfying the condition in Theorem 5.3.5. Let $\mathbb{G}'$ be a strongly connected and balanced graph with n disconnected components and the weights of the edges in the graph are positive. Let L' be its Laplacian matrix. Multiplying from the left a positive definite matrix S gives us a Laplacian matrix $L = SL'$ satisfying the condition in Theorem 5.3.5 [1].

5.4 Illustrative examples

In this section, several examples are given to illustrate the theoretical analysis results.

5.4.1. Example. (*Clustering with different self-dynamics*) Consider the network consisting of two clusters of informed agents and one cluster of naive agents with $l_1 = l_2 = l_3 = 2$ and $b_1 = 1,\ b_2 = 7$. The Laplacian matrix is given by

$$G = \left[\begin{array}{cc|cc|cc} 2 & 0 & -1 & -1 & 0 & 0 \\ 0 & 2 & -2 & 0 & 0 & 0 \\ \hline -1 & 0 & 1 & 0 & 0 & 0 \\ 0 & -1 & 0 & 1 & 0 & 0 \\ \hline -1 & 0 & -1 & -1 & 3 & 0 \\ 0 & -2 & -4 & 0 & 0 & 6 \end{array}\right].$$

Since the final values of the first and second clusters are 4 and 5.5, respectively, the values of the naive agents converge to $4 \times \frac{1}{3} + 5.5 \times \frac{2}{3} = 5$. Fig. 5.1 shows the evolution of the three clusters.

5.4.2. Example. (*Delay-induced cluster synchronization*) Let

$$L_1 = \left[\begin{array}{cc|cc} 3 & -2 & 0 & -1 \\ -2 & 3 & -1 & 0 \\ \hline -1 & 0 & 2 & -1 \\ 0 & -1 & -1 & 2 \end{array}\right], \quad L_2 = \left[\begin{array}{cc|cc|cc} 4 & -2 & 0 & -1 & 0 & -1 \\ -2 & 4 & -1 & 0 & -1 & 0 \\ \hline -1 & 0 & 3 & -1 & -1 & 0 \\ 0 & -1 & -1 & 3 & 0 & -1 \\ \hline 0 & -1 & -1 & 0 & 2 & 0 \\ -1 & 0 & 0 & -1 & 0 & 2 \end{array}\right].$$

All the agents in the coupled network (5.16) have the same self-dynamics, which are [114]

$$\begin{aligned} \dot{x}_i(t) \quad = \quad & \begin{bmatrix} -3.6 & 0 \\ 0 & -4.2 \end{bmatrix} \begin{bmatrix} x_{i1}(t) \\ x_{i2}(t) \end{bmatrix} + \begin{bmatrix} a\cos(\nu t) \\ 0 \end{bmatrix} \\ & + \begin{bmatrix} 1.5 & -0.5 \\ -2.1 & 1.8 \end{bmatrix} \begin{bmatrix} \frac{1}{2}(|x_{i1}(t)+1| - |x_{i1} - 1|) \\ \frac{1}{2}(|x_{i2}(t)+1| - |x_{i2} - 1|) \end{bmatrix}. \end{aligned} \tag{5.28}$$

[1] We are indebted to I. Shames for pointing out this reformulation of some of our earlier results.

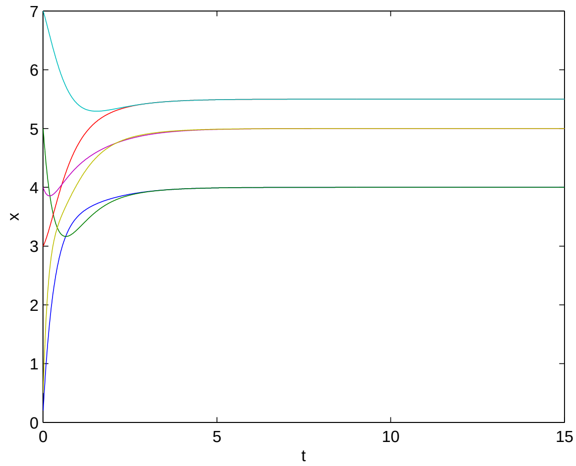
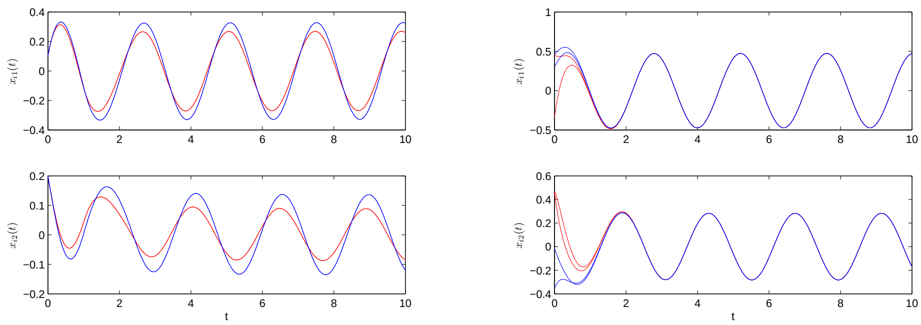


Figure 5.1: *The evolution of a system consisting of three clusters.*



(a) When $\tau = 1$, the agents evolve into 2-clusters.

(b) When $\tau = 0$, the agents achieve complete synchronization.

Figure 5.2: *The evolution of the states $x_i(t)$ for $i = 1, \dots, 4$.*

When $a = 1.6$ and $\nu = 2.6$, system (5.28) has a unique and globally exponentially stable periodic solution.

Consider the coupled network associated with the coupling matrix L_1. Let $\tau = 1$, $c = 0.5$ and $\Gamma = diag\{1, 1\}$. Using Matlab, we get solutions Q_j and P_j to (5.20) and (5.22) as $Q_j = P_j = diag\{0.5550, 0.5550, 0.4717, 0.4717\}$, $j = 1, \dots, m$. Assume that every agent takes the same initial value $x_i(\theta) = [0.1, 0.2]^T$, $i = 1, \dots, 4$, $\theta \in [-1, 0]$.

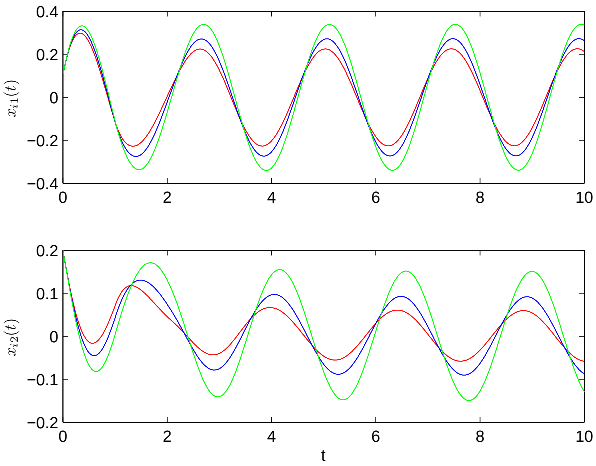


Figure 5.3: *The agents evolve into 3-clusters with L_2 when $\tau = 1$.*

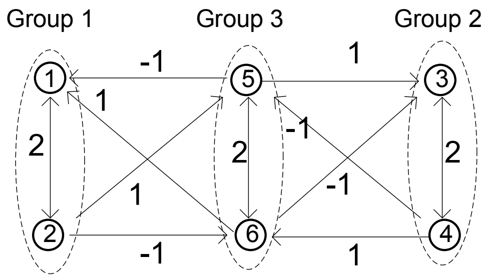


Figure 5.4: *The topology of a network.*

The states of the agents finally evolve into two clusters, as shown in Fig. 5.2(a). When $\tau = 0$, the states of the agents achieve complete synchronization as shown in Fig. 5.2(b). So the delay indeed has induced the clustering behavior in this example.

When the coupled network corresponds to the Laplacian matrix L_2, and $\tau = 1$, from Fig. 5.3 it can be seen that the agents finally evolve into three clusters.

5.4.3. Example. (*Clustering with negative couplings*) A network that realizes 2-cluster synchronization has the topology shown in Fig. 5.4. The Laplacian matrix L

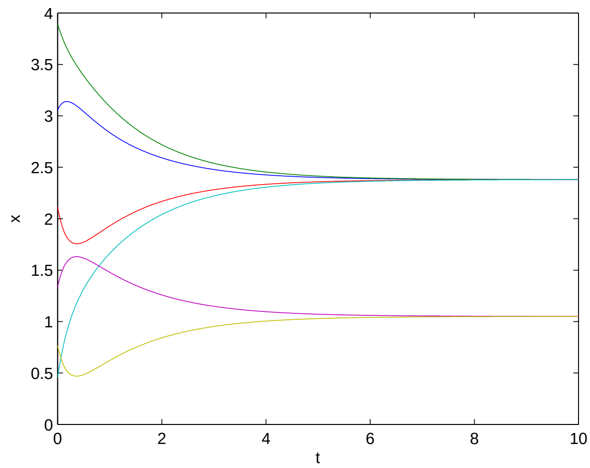


Figure 5.5: *State trajectories. (Agents 1,2,3,4 are in the same cluster)*

is

$$\left[\begin{array}{cccc|cc} 2 & -2 & 0 & 0 & 1 & -1 \\ -2 & 2 & 0 & 0 & 0 & 0 \\ 0 & 0 & 2 & -2 & -1 & 1 \\ 0 & 0 & -2 & 2 & 0 & 0 \\ \hline 0 & -1 & 0 & 1 & 2 & -2 \\ 0 & 1 & 0 & -1 & -2 & 2 \end{array}\right],$$

which has two zero eigenvalues and the other eigenvalues have positive real parts. Let groups 1, 2, 3 be $\{1,2\}$, $\{3,4\}$, $\{5,6\}$, respectively. It is easy to see from Fig. 5.4 and Fig. 5.5 that, although there is no direct connection between groups 1 and 2, the states of the agents in these two groups finally achieve the same value via the interconnection with agents in group 3, which have a different final value.

5.5 Conclusion

This chapter has investigated three different algorithms that lead to n-cluster synchronization in multi-agent systems. Some sufficient conditions and/or necessary conditions have been constructed for systems with different agent self-dynamics, with delay or having negative couplings. Numerical examples are given to verify the effectiveness of the analysis. The three approaches presented here are just examples of different approaches towards cluster synchronization. It is envisioned that after gaining insights into the clustering behavior in natural, social or engineered systems, more approaches can be revealed and thus different cluster synchronization models can be constructed whose advantages and disadvantages can be compared.

Chapter 6

Controllability and cluster synchronization of multi-agent systems

Controllability is an essential property of a dynamical system that plays a critical role in control design problems. Synchronization phenomena have been widely observed in natural and man-made systems and have attracted significant attention from researchers in several disciplines, including statistical physics, computer science, network science as well as electrical engineering. There are of course differences between the controllability problem and the cluster synchronization problem for complex multi-agent systems. For example, cluster synchronization, or synchronization in general, is concerned with a complex network's collective asymptotic behavior when time approaches infinity; in comparison, controllability is concerned with a system's dynamic behavior within finite time. However, in this chapter, by positioning the two problems together, deeper insight can be gained into both of the two topics. Towards this end, we first define generalized equitable partitions and almost equitable partitions for general directed weighted graphs. Then we are able to provide an upper bound and a lower bound for the controllable subspace for a general diffusively coupled multi-agent system. We point out the close relationship between the generalized almost equitable partition of a graph and the constant-row-sums property of the block sub-matrices of the graph's Laplacian matrix. Furthermore, we show that diffusively coupled multi-agent networks that are not controllable tend to realize cluster synchronization.

6.1 Controllability of multi-agent systems

We consider a multi-agent system consisting of N agents and we use $\mathcal{N} = \{1, \ldots, N\}$ to denote the set of indices of all the agents. Let $x_i \in \mathbb{R}$, $i \in \mathcal{N}$, denote the state of agent i. We assign the roles of the *leaders* and *followers* to the agents and use $\mathcal{N}_L$, $\mathcal{N}_F \subseteq \mathcal{N}$ to denote the sets of indices of the leaders and followers, respectively. Assume that there are altogether $0 < s = |\mathcal{N}_L| < N$ control inputs $u_i \in \mathbb{R}$, $1 \leqslant i \leqslant s$ and each leader is influenced by only one input. For a leader $i \in \mathcal{N}_L$, let $[i] \in \{1, \ldots, s\}$ denote the index of the control input acting on it. Let $\mathbb{G} = (\mathcal{V}, \mathcal{E})$ with the vertex set $\mathcal{V} = \{v_1, v_2, \ldots, v_N\}$ be a directed weighted graph that represents the interaction topology among the agents. Let A and L be the corresponding adjacency matrix and Laplacian matrix of $\mathbb{G}$ and let d_i^{in} be the in-degree of node i and $D^{in} = diag\{d_1^{in}, \ldots, d_N^{in}\}$. Then the dynamics of the leaders are determined by

$$\dot{x}_i = \sum_{j=1}^{N} a_{ij}(x_j - x_i) + u_{[i]}, \quad i \in \mathcal{N}_L, \tag{6.1}$$

and the followers' dynamics are governed by linear diffusive couplings

$$\dot{x}_i = \sum_{j=1}^{N} a_{ij}(x_j - x_i), \quad i \in \mathcal{N}_F, \tag{6.2}$$

where a_{ij} is the ijth element of A satisfying $a_{ij} \geqslant 0$, for $j \neq i$.

Let $x = [x_1, \ldots, x_N]^T$ and $u = [u_1, \ldots, u_s]^T$. Then (6.1) and (6.2) can be written in a compact form

$$\dot{x} = -Lx + Mu, \tag{6.3}$$

where the elements of M are defined by

$$m_{ij} = \begin{cases} 1 & \text{if } j = [i] \\ 0 & \text{otherwise,} \end{cases}$$

for $1 \leqslant i \leqslant N$ and $1 \leqslant j \leqslant s$.

Controllability is a classical notion in control theory and a dynamical system is said to be *controllable* if under suitable control actions as the system's inputs, the system's state can be driven from any initial values to any desired final values within finite time [49]. The controllability problem of system (6.3) has attracted great attention from the area of systems and control [66, 113]. Denote the controllable subspace of system (6.3) by $\mathcal{R}$. Note that $\mathcal{R}$ is the smallest L-invariant subspace that contains the subspace spanned by the columns of M, denoted by $im(M)$ [113]. In order to characterize the controllable subspace, we need some more notions from graph theory.

6.2 Controllability through generalized almost equitable partitions

Given a partition $\pi = \{C_1, C_2, \ldots, C_n\}$ of the node set $\mathcal{V} = \{v_1, \ldots, v_N\}$ of a graph $\mathbb{G} = (\mathcal{V}, \mathcal{E})$, we call C_i's the cells and n the size of the partition. Let $|C_i| = l_i$. We can always relabel the nodes such that the first l_1 nodes lie in C_1, the next l_2 nodes lie in C_2, and so on. Then we can write the adjacency matrix A and the Laplacian matrix L in the following block matrix form according to the partition

$$A = \begin{bmatrix} A_{11} & A_{12} & \cdots & A_{1n} \\ A_{21} & A_{22} & \cdots & A_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ A_{n1} & A_{n2} & \cdots & A_{nn} \end{bmatrix}, \tag{6.4}$$

$$L = \begin{bmatrix} L_{11} & L_{12} & \cdots & L_{1n} \\ L_{21} & L_{22} & \cdots & L_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ L_{n1} & L_{n2} & \cdots & L_{nn} \end{bmatrix}, \tag{6.5}$$

where $A_{ij}, L_{ij} \in \mathbb{R}^{l_i \times l_j}$, $1 \leqslant i, j \leqslant n$. The *characteristic matrix* $P(\pi) \in \mathbb{R}^{N\times n}$ of the partition is defined by

$$P_{ij}(\pi) = \begin{cases} 1 & \text{if } v_i \in C_j \\ 0 & \text{otherwise,} \end{cases}$$

for $1 \leqslant i \leqslant N$ and $1 \leqslant j \leqslant n$.

First consider the case when the graph $\mathbb{G} = (\mathcal{V}, \mathcal{E})$ is unweighted and undirected as in [66, 113], meaning that for any distinct pair of nodes v_i and v_j, if $(v_i, v_j) \in \mathcal{E}$, then $(v_j, v_i) \in \mathcal{E}$ and the weights $a_{ij} = a_{ji} = 1$. We say agent j is a neighbor of agent i, if $a_{ij} = 1$. A partition π is said to be an *equitable partition* if each node in C_j has the same number of neighbors in C_i for all $1 \leqslant i, j \leqslant n$. If one only cares about the number of neighbors in adjacent cells, while ignoring the structure inside a cell, one can define the notion of almost equitable partition. A partition π is said to be an *almost equitable partition* if each node in C_j has the same number of neighbors in C_i for all $1 \leqslant i, j \leqslant n$ and $i \neq j$.

However, when we consider general directed weighted graphs, the weights a_{ij} can have any nonnegative value. Thus we cannot employ the notion of the number of neighbors any more. Now we generalize the notions of equitable partitions and almost equitable partitions [39] in a natural way.

6.2.1. Definition. *A partition π is said to be a generalized equitable partition if for any $v_k, v_l \in C_i$, $i, j = 1, \ldots, n$,*

$$\sum_{v_r \in C_j} a_{kr} = \sum_{v_r \in C_j} a_{lr}. \tag{6.6}$$

6.2.2. Definition. *A partition π is said to be a generalized almost equitable partition if for any $v_k, v_l \in C_i$, $i \neq j$, $i, j = 1, \ldots, n$,*

$$\sum_{v_r \in C_j} a_{kr} = \sum_{v_r \in C_j} a_{lr}. \tag{6.7}$$

From the above two definitions, we can see the close relationships between the generalized equitable partitions (resp. almost equitable partitions) of a graph and the constant-row-sums property of block matrices A_{ij} (resp. L_{ij}) of the associated adjacency matrix A (resp. Laplacian matrix L).

6.2.1. PROPOSITION. *For a partition π of a graph $\mathbb{G}$, we always label the nodes such that the first l_1 nodes lie in C_1, the next l_2 nodes lie in C_2, and so on. A partition π is a generalized equitable partition of a graph $\mathbb{G}$ if and only if the row sums of each block A_{ij} of the associated adjacency matrix A written in form (6.4) are equal. A partition π is a generalized almost equitable partition if and only if the row sums of each block L_{ij} of the associated Laplacian matrix L written in form (6.5) are equal.*

The *quotient graph* of $\mathbb{G}$ with respect to a generalized almost equitable partition π, denoted by $\mathbb{G}/\pi$, is a directed weighted graph whose node set is $\mathcal{V}(\mathbb{G}/\pi) = \pi$, the edge set is the set of ordered pairs such that (C_i, C_j) is an edge of $\mathbb{G}/\pi$ if and only if $i \neq j$ and there exist v_i in C_i and v_j in C_j such that $(v_i, v_j) \in \mathcal{E}(G)$ and the weight associated with each edge (C_i, C_j) of $\mathbb{G}/\pi$ is $a^{\pi}_{ji} = \sum_{v_k \in C_i} a_{jk}$. Let A^{π} and L^{π} be the adjacency and Laplacian matrices of $\mathbb{G}/\pi$, respectively.

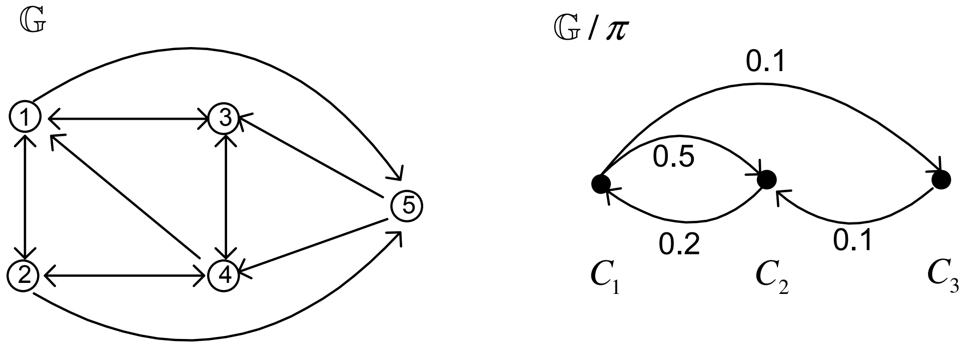


Figure 6.1: *A directed weighted graph $\mathbb{G}$ and its quotient graph $\mathbb{G}/\pi$.*

6.2.1. EXAMPLE. Let the adjacency matrix A and the Laplacian matrix L associated with a graph $\mathbb{G}$ shown in Fig. 6.1 be given by

$$A = (A_{ij})_{3\times 3} = \left[\begin{array}{cc|cc|c} 0 & 2 & 0.1 & 0.1 & 0 \\ 0.1 & 0 & 0 & 0.2 & 0 \\ \hline 0.5 & 0 & 0 & 0.5 & 0.1 \\ 0 & 0.5 & 0.5 & 0 & 0.1 \\ \hline 0.1 & 0.1 & 0 & 0 & 0 \end{array}\right],$$

$$L = (L_{ij})_{3\times 3} = \left[\begin{array}{cc|cc|c} 2.2 & -2 & -0.1 & -0.1 & 0 \\ -0.1 & 0.3 & 0 & -0.2 & 0 \\ \hline -0.5 & 0 & 1.1 & -0.5 & -0.1 \\ 0 & -0.5 & -0.5 & 1.1 & -0.1 \\ \hline -0.1 & -0.1 & 0 & 0 & 0.2 \end{array}\right].$$

It is easy to see that $L_{ij},\ i, j = 1, 2, 3$, have constant row sums, which corresponds to a nontrivial generalized almost equitable partition $\pi = \{C_1 = \{v_1, v_2\}, C_2 = \{v_3, v_4\}, C_3 = \{v_5\}\}$ of $\mathbb{G}$. Note that π is not a generalized equitable partition since the row sums of A_{11} are not equal. Then the characteristic matrix $P(\pi)$ of the partition π is

$$P(\pi) = \begin{bmatrix} 1 & 0 & 0 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}.$$

The quotient graph $\mathbb{G}/\pi$ of $\mathbb{G}$ with respect to π is also shown in Fig. 6.1 and its adjacency and Laplacian matrices are

$$A^{\pi} = \begin{bmatrix} 0 & 0.2 & 0 \\ 0.5 & 0 & 0.1 \\ 0.2 & 0 & 0 \end{bmatrix},\ L^{\pi} = \begin{bmatrix} 0.2 & -0.2 & 0 \\ -0.5 & 0.6 & -0.1 \\ -0.2 & 0 & 0.2 \end{bmatrix}.$$

$\square$

For given nodes $v_1, \ldots, v_s \in \mathcal{V}$, π is said to be a *generalized almost equitable partition relative to* $v_1, \ldots, v_s$ if it is a generalized almost equitable partition and $\{v_1\}, \ldots, \{v_s\}$ are its cells. Let Π_{GEP}, Π_{GAEP} and $\Pi_{GAEP}(v_1, \ldots, v_s)$ denote the sets of all generalized equitable, generalized almost equitable and generalized almost equitable partitions relative to $v_1, \ldots, v_s$, respectively. Moreover, we say that a generalized almost equitable partition relative to $v_1, \ldots, v_s$ is maximal, which is denoted by $\pi^*_{GAEP}(v_1, \ldots, v_s)$ if it has the smallest size; that is, if it contains the fewest possible cells. It can be shown that given a graph $\mathbb{G}$ and nodes $v_1, \ldots, v_s$, $\pi^*_{GAEP}(v_1, \ldots, v_s)$ always exists uniquely.

Another class of partitions depends on the distance between two vertices and the graph diameter. Given a strongly connected graph $\mathbb{G}$, a partition $\pi = \{\{u \in \mathcal{V} | dist(v,u) = i\},\ i = 0, 1, \ldots, diam(\mathbb{G})\}$ is called the *distance partition relative to* v. The distance partition relative to a vertex v is unique and denoted by $\pi_D(v)$. Let C_{i+1} be the cell $\{u \in \mathcal{V} | dist(v,u) = i\}$ in $\pi_D(v)$, where $0 \leqslant i \leqslant diam(\mathbb{G})$.

6.3 Controllability and cluster synchronization of multi-agent systems

Now we are ready to use the graph notions introduced in the previous section to derive lower and upper bounds for the controllable subspace of system (6.3). The following result is proved in [20] for undirected unweighted graphs and the necessary part is restated in [38] for undirected weighted graphs. It is also valid for directed weighted graphs with respect to generalized almost equitable partitions.

6.3.1. Lemma. *Let $\mathbb{G} = (\mathcal{V}, \mathcal{E})$ be a directed weighted graph, let L be its Laplacian matrix, let $\pi = \{C_1, \ldots, C_n\}$ be a partition of $\mathcal{V}$, and let $P(\pi)$ be the characteristic matrix of π. Then π is a generalized almost equitable partition whose cardinality equals n if and only if there is an $n \times n$ matrix B such that*

$$LP(\pi) = P(\pi)B. \tag{6.8}$$

If π is a generalized almost equitable partition, then B is the Laplacian matrix L^π of the quotient graph $\mathbb{G}/\pi$.

An immediate consequence of this Lemma is that we can characterize the generalized almost equitable partitions using terms of invariant subspaces of the Laplacian matrix L of $\mathbb{G}$ as follows.

6.3.2. Lemma. *A partition π of $\mathbb{G}$ is a generalized almost equitable partition, if and only if $im(P(\pi))$ is L-invariant.*

An upper bound for the controllable subspace is given in [113] for system (6.3) with multiple leaders when the graph is undirected and unweighted. We provide an upper bound in terms of generalized almost equitable partitions for the general case when the graph is directed and weighted.

6.3.1. Proposition. *Let $\mathcal{V}_L = \{v_1, \ldots, v_s\}$, and $\pi \in \Pi_{GAEP}(v_1, \ldots, v_s)$. Then*

$$\mathcal{R} \subseteq im(P(\pi)),$$

and

$$dim(\mathcal{R}) \leqslant |\pi^*_{GAEP}(v_1, \ldots, v_s)|.$$

Proof. It follows from Lemma 6.3.2 that $im(P(\pi))$ is L-invariant. Since $\mathcal{V}_L = \{v_1, \ldots, v_s\}$, and $\pi \in \Pi_{GAEP}(v_1, \ldots, v_s)$, we have $im(M) \subseteq im(P(\pi))$; that is $im(P(\pi))$ is an L-invariant subspace containing $im(M)$. In view of the fact that $\mathcal{R}$ is the smallest L-invariant subspace that contains the subspace spanned by the columns of M, it follows that $\mathcal{R} \subseteq im(P(\pi))$. Thus

$$dim(\mathcal{R}) \leqslant dim(im(P(\pi^*_{GAEP}(v_1, \ldots, v_s)))) = |\pi^*_{GAEP}(v_1, \ldots, v_s)|.$$

□

From this proposition, we immediately have the following result.

6.3.2. Proposition. *Assume that $\mathbb{G}$ is strongly connected. System (6.3) with multiple leaders $\mathcal{V}_L = \{v_1, \ldots, v_s\}$, is controllable only if $\pi^*_{GAEP}(v_1, \ldots, v_s)$ is trivial; that is, $\{v_i\} \in \pi^*_{GAEP}(v_1, \ldots, v_s)$ for all $v_i \in \mathcal{V}$.*

For system (6.3) with a single leader, we have the following lower bound in terms of the distance partitions.

6.3.3. Proposition. *If $\mathcal{V}_L = \{v\}$, then*

$$|\pi_D(v)| \leqslant dim(\mathcal{R}).$$

Proof. Without loss of generality, assume $v = v_1$, $\pi_D(v_1) = \{C_1, \ldots, C_n\}$ and $|C_i| = l_i$, where $1 \leqslant n \leqslant diam(\mathbb{G}) + 1$, $C_1 = \{v_1\}$, and $C_{i+1} = \{v_{l_1+\cdots+l_i+1}, \ldots, v_{l_1+\cdots+l_{i+1}}\}$, $1 \leqslant i \leqslant n-1$. The Laplacian matrix L can be written in the following form.

$$L = \begin{bmatrix} d_1^{in} & * & * & \cdots & * & * \\ L_{21} & L_{22} & * & \cdots & * & * \\ \mathbf{0} & L_{32} & L_{33} & \cdots & * & * \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ \mathbf{0} & O & O & \cdots & L_{n-1,n-1} & * \\ \mathbf{0} & O & O & \cdots & L_{n,n-1} & L_{nn} \end{bmatrix},$$

where "$*$" represents the entry in the matrix which is irrelevant to the present discussion. Since $\mathcal{V}_L = \{v\}$, the matrix M is an N-dimensional column vector $e_1 = [1, 0, \ldots, 0]^T$.

Let $E = [e_1 \ Le_1 \ \cdots L^{n-1}e_1]$. Some calculations give that

$$E = \begin{bmatrix} 1 & * & * & \cdots & * \\ \mathbf{0} & L_{21} & * & \cdots & * \\ \mathbf{0} & O & L_{32}L_{21} & \cdots & * \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ \mathbf{0} & O & O & \cdots & L_{n,n-1}\cdots L_{21} \end{bmatrix}.$$

Since the graph $\mathbb{G}$ is strongly connected and $C_{i+1} = \{u \in \mathcal{V} | dist(v, u) = i\}$, there exists a directed path of length i from v_1 to every node in C_{i+1}, $1 \leqslant i \leqslant n-1$. This implies that the matrix product $L_{i+1,i} \cdots L_{21}$, $1 \leqslant i \leqslant n-1$, contains a positive entry in each row. Then $\operatorname{rank}(E) = |\pi_D(v)| = n$. Thus we can conclude that

$$|\pi_D(v)| = \operatorname{rank}(E) \leqslant \operatorname{rank}([e_1 \ Le_1 \ \cdots L^{N-1}e_1]) = dim(\mathcal{R}).$$

□

Combining Propositions 6.3.1 and 6.3.3, we can obtain the following lower and upper bounds for the dimension of the controllable subspace of system (6.3) with a single leader.

6.3.4. Proposition. *If $\mathcal{V}_L = \{v\}$, then*

$$|\pi_D(v)| \leqslant dim(\mathcal{R}) \leqslant |\pi^*_{GAEP}(v)|.$$

Note that the bounds given here are tight and cannot be improved further, which can be seen from examples in [113].

Let $\mathcal{V}_L = \{v_1, \ldots, v_s\}$, let $\{C_1, C_2, \ldots, C_n\}$ be a partition of $\mathcal{V}$ and $\{v_i\}$ are cells of the partition $\{C_1, C_2, \ldots, C_n\}$ for all $i = 1, \ldots, s$. Define the manifold

$$\mathcal{X} = \{x = [x_1, \ldots, x_N]^T | \ x_{i_1}(t) = x_{i_2}(t), \text{ for all } v_{i_1}, v_{i_2} \in C_i, \ i = 1, \ldots, n\}, \quad (6.9)$$

which corresponds to the n-cluster synchronization defined in Chapter 5. The following result reveals the relationship between the invariance of the cluster synchronization manifold and the existence of generalized almost equitable partitions.

6.3.1. Theorem. *The n-cluster synchronization manifold $\mathcal{X}$ of system (6.3) is invariant for any input $u(t)$ if and only if the partition $\{C_1, C_2, \ldots, C_n\}$ is a generalized almost equitable partition. In addition, if the graph $\mathbb{G}$ is strongly connected and $\{C_1, C_2, \ldots, C_n\}$ is a generalized almost equitable partition, then the manifold $\mathcal{X}$ is asymptotically stable, i.e., $\lim_{t\to\infty} x_{i_1}(t) - x_{i_2}(t) = 0$, for all $v_{i_1}, v_{i_2} \in C_i$, $1 \leqslant i \leqslant n$.*

Proof. The necessary and sufficient condition can be proved by using similar arguments to the proof of Theorem 3.11 in [58]. Next we prove the asymptotic stability of the manifold $\mathcal{X}$. The solution to system (6.3) is given by

$$x(t) = e^{-Lt}x(0) + \int_0^t e^{-L(t-s)}Mu(s)ds.$$

Since $\{C_1, C_2, \ldots, C_n\}$ is a generalized almost equitable partition, the corresponding block matrices L_{ij} have constant row sums. If we partition the matrix $e^{-L(t-s)}$ to $n \times n$ blocks for any $t \geqslant s \geqslant 0$ with respect to the partition $\{C_1, C_2, \ldots, C_n\}$ as in (6.5), then each block matrix $(e^{-L(t-s)})_{ij}$ also has constant row sum. Therefore, $\int_0^t e^{-L(t-s)}Mu(s)ds \in \mathcal{X}$ for all $t \geqslant 0$. Since the graph $\mathbb{G}$ is strongly connected, 0 is a simple eigenvalue of the matrix $-L$ and all the other eigenvalues of $-L$ have negative real parts. It follows from Lemma 5.3.2 that

$$\lim_{t\to\infty} e^{-Lt} = \eta_1 \alpha_1^T,$$

where η_1 and α_1 are the right and left eigenvectors corresponding to the eigenvalue 0 of $-L$ satisfying that $\alpha_1^T \eta_1 = 1$. η_1 can be simply taken as $\eta_1 = \mathbf{1}_N$. Thus $\eta_1 \alpha_1^T$ is a matrix with all rows the same. Suppose that $v_{i_1}, v_{i_2} \in C_i,\ 1 \leqslant i \leqslant n$. Then one can obtain that

$$\lim_{t\to\infty} x_{i_1}(t) - x_{i_2}(t) = \lim_{t\to\infty} (e^{-Lt}x(0))_{i_1} - (e^{-Lt}x(0))_{i_2} = 0,$$

which proves the asymptotic stability of the manifold $\mathcal{X}$. □

6.3.1. Remark. *In view of the definition of cluster synchronization given in Definition 5.1.1, Theorem 6.3.1 has proved that the agents will finally form n clusters and synchronize within the clusters, while the differences between clusters may or may not converge to zero. This is exactly the "group consensus" we discussed in Remark 5.1.1, which is weaker than the cluster synchronization defined in Definition 5.1.1.*

Combining Proposition 6.3.2 and Theorem 6.3.1, we have the following result.

6.3.2. Theorem. *Let $\mathcal{V}_L = \{v_1, \ldots, v_s\}$. Assume that the graph $\mathbb{G}$ is strongly connected and it has a nontrivial $\pi^*_{GAEP}(v_1, \ldots, v_s)$. System (6.3) is not controllable and it realizes group consensus with respect to the partition $\pi^*_{GAEP}(v_1, \ldots, v_s)$.*

In this section, by comparing the conditions for realizing cluster synchronization and checking controllability, we have gained the insight that those multi-agent networks that are uncontrollable in finite time tend to realize cluster synchronization as time goes to infinity.

6.4 Illustrative example

We take the graph $\mathbb{G}$ and the associated Laplacian matrix L in Example 6.2.1 as an example. If we take node 5 as a leader, then the maximal generalized almost equitable partition relative to node 5 is $\pi^*(5) = \{\{1,2\},\{3,4\},\{5\}\}$. System (6.3) with this Laplacian matrix is uncontrollable since the partition $\pi^*(5)$ is nontrivial. Assume that the initial values of the agents are chosen randomly from $[0, 10]$ and the control input $u(t) = cos(2\pi t)$. From Theorem 6.3.1, we know that system (6.3) will finally evolve into three clusters with respect to the partition $\pi^*(5)$ as shown in Fig. 6.2.

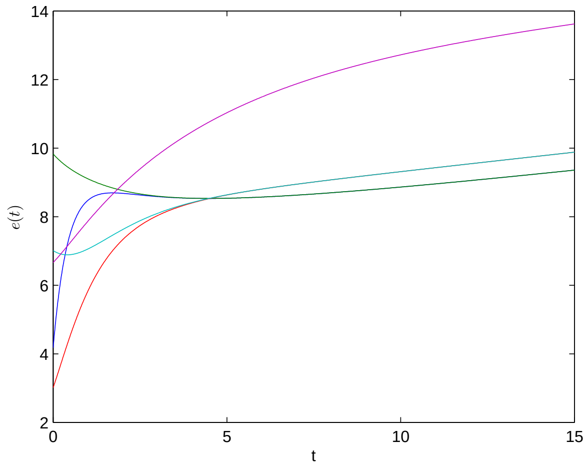


Figure 6.2: *The five agents evolve into three clusters.*

6.5 Conclusion

We have looked at jointly the controllability problem and the cluster synchronization problem for multi-agent systems. Using the notions of generalized graph partitioning, we have provided upper and lower bounds for the controllable subspace of a diffusively coupled multi-agent system. We have also gained insight that the multi-agent networks that are uncontrollable in finite time tend to realize cluster synchronization as time goes to infinity. Illustrative example has verified the effectiveness of the theoretical results.

Chapter 7

Determination of clock synchronization errors in distributed networks

As physical devices, such as computational units, sensors and actuators, are more and more frequently working together over distances, people are more and more concerned with the problem of how to synchronize the clocks that are installed at those physical devices and connected through wired and/or wireless data networks [36]. The convergence analysis of a synchronous algorithm in Chapter 3 has also illustrated the importance of clock synchronization in a network, where the converging algorithm may not converge any more when the clock installed at each agent is not synchronized with each other and the agents can only update according to their own clocks. Recently, it has been shown by Freris, Graham and Kumar [37] that clocks in distributed networks cannot be synchronized precisely in the presence of asymmetric time delays even in idealized situations. Motivated by that impossibility result, we test under similar settings the performance of some existing clock synchronization protocols and show that the synchronization errors can be bounded within an acceptable level of accuracy that are determined by the degree of asymmetry in time delays. After studying the basic case of synchronizing two clocks in the two-way message passing process, we analyze directed ring networks, in which neighboring clocks are likely to experience severe asymmetric time delays. We then discuss connected undirected networks with two-way message passing between each pair of adjacent nodes. In the end, we expand the discussions to networks with directed topologies that are strongly connected.

7.1 Models for clocks in networks

As in [37], we consider affine models for clocks. Let $i > 0$ be the label of a clock in a network, and denote its display by x_i. Then the evolution of x_i can be described by

$$x_i(t) = a_i t + b_i, \tag{7.1}$$

where t is the time of a standard reference clock, $a_i > 0$ is called the *skew* that is the ratio of the speed of clock i with respect to the reference clock, and b_i is called the *offset* that is the difference between the display of clock i and the reference clock at time $t = 0$. Here we consider the idealized case when the skews of the clocks are fixed, e.g. not affected by the changes in the environmental temperature, and the communications within the network are noiseless and fault-free.

Assume that the clocks are installed at nodes in a distributed network. We use the label of the clock to denote the node where the clock is installed. It is assumed in [37] that when a message is sent from node i to another node j, the latter can only receive it after a fixed but unknown time delay $d_{ij} > 0$. In addition, the time delays are not necessarily symmetric, and in fact for a pair of distinct nodes i and j, d_{ij} is in general not equal to d_{ji}. In order to describe the message passing process between clocks more conveniently, in this chapter, we use i instead of v_i to denote a node and use $\mathcal{V} = \{1, \ldots, N\}$ instead of $\mathcal{V} = \{v_1, \ldots, v_N\}$ to denote the node set, which differ from the notation used in previous chapters. In what follows, we will use a graph $\mathbb{G}$ with the node set $\mathcal{V} = \{1, \ldots, N\}$ and the edge set $\mathcal{E} \subset \{(i, j) : i, j \in \mathcal{V}\}$ to describe the topology of a network consisting of N nodes. In $\mathbb{G}$, there is a directed edge from node i to j if i can send messages to j; correspondingly, there is an undirected edge between i and j if both i and j can send messages to each other.

7.2 Synchronizing two clocks

In this section, we consider two clock synchronization. For analysis purposes, we can always describe the message passing process with respect to the standard reference clock. In the sequel, we use the sequence $\{t_k\}$, $k \geqslant 0$, to denote the set of time instants embedded in the reference time axis t, at which a clock sends or receives messages. Then the message exchange process for two clocks 1 and 2 trying to get synchronized is illustrated in Figure 7.1. At time t_0, node 1 sends a message of its current value of $x_1(t_0)$ to node 2. We say node 1 has sent a message *time stamped* by its clock just before the transmission. Node 2 records the time $x_2(t_1)$ when it receives the message $x_1(t_0)$ and after a constant time w_1, it sends the message $x_2(t_1)$ at the time t_2 back to node 1 with the time-stamp $x_2(t_2)$. Correspondingly, node 1 receives this message at time t_3 and records the time $x_1(t_3)$. It then sends a message after a constant time w_2. In this manner the messages are sent back and forth.

Without loss of generality, take the skew of clock 1 to be 1, i.e. $a_1 = 1$. As shown in [37], the skew a_2 of clock 2 and the round-trip delay $d_{12} + d_{21}$ can be calculated precisely by

$$a_2 = \frac{x_2(t_5) - x_2(t_1)}{x_1(t_4) - x_1(t_0)}, \tag{7.2}$$

$$d_{12} + d_{21} = x_1(t_3) - x_1(t_0) - \frac{1}{a_2}(x_2(t_2) - x_2(t_1)). \tag{7.3}$$

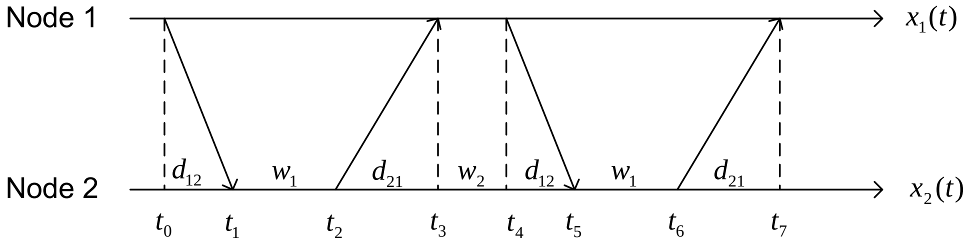


Figure 7.1: *Message exchanges between two clocks.*

However, the individual time delays d_{12} and d_{21} can *never* be determined precisely when they are asymmetric and this is part of the synchronization impossibility result for a pair of clocks shown in [37], which as argued in the same paper leads to synchronization errors that cannot be eliminated. We also refer the interested reader to [48] for more information about phase and skew estimators.

Now we try to synchronize the two clocks by repeatedly updating their displays. Consider first the simple case when $a_2 = 1$ as well; in other words, the skews of the two clocks are the same. We use D to denote the round-trip time delay $d_{12} + d_{21} = x_1(t_3) - x_1(t_0) - (x_2(t_2) - x_2(t_1))$. When the two clocks update their displays, they use the average delay $\bar{D} = \frac{D}{2}$ as the *nominal* delay to compensate the time-stamped messages they receive about the most recent values of the other clock's display. For example, when clock 1 receives a message of $x_2(t_k)$ from clock 2, it takes $x_2(t_k) + \bar{D}$ as the estimated current value of the display of clock 2. The same estimation strategy is adopted by both of the two clocks. To get synchronized, after a clock receives a new message from the other, it always updates its display to the average of its current display and the latest estimation of the other clock's current display. We assume the updates take place instantaneously and the message exchanges are carried out repeatedly.

The embedding technique to write down a distributed system's dynamics with respect to a common reference time axis for analysis purposes has been used before when studying distributed and parallel computations and asynchronous systems [10, 15]. Following this approach, we use the sequence $\{t_k\}$, $k \geqslant 0$, embedded in the reference time axis t, to write the system equations. Although the two clocks update periodically according to their own clocks, since the clocks have the same skew, we know that for any time $\tau > 0$, there always exists $k \geqslant 0$ such that $t_k \leqslant \tau < t_{k+1}$ and $x_1(\tau) - x_2(\tau) = x_1(t_k) - x_2(t_k)$. For the sake of conciseness, in this chapter we use the notation $x_i(k)$ instead of $x_i(t_k)$. Then the system equations of the updating process of the two clocks after embedding can be written as

$$\begin{cases} x_1(4k+1) = x_1(4k) + d_{12} \\ x_2(4k+1) = \frac{1}{2}((x_1(4k) + \bar{D}) + (x_2(4k) + d_{12})) \end{cases}$$
$$\begin{cases} x_1(4k+2) = x_1(4k+1) + l_1 d_{12} \\ x_2(4k+2) = x_2(4k+1) + l_1 d_{12} \end{cases}$$
$$\begin{cases} x_1(4k+3) = \frac{1}{2}((x_2(4k+2) + \bar{D}) + (x_1(4k+2) + d_{21})) \\ x_2(4k+3) = x_2(4k+2) + d_{21} \end{cases} \tag{7.4}$$
$$\begin{cases} x_1(4(k+1)) = x_1(4k+3) + l_2 d_{12} \\ x_2(4(k+1)) = x_2(4k+3) + l_2 d_{12}, \end{cases}$$

where $k \geqslant 0$ and $l_i = \frac{w_i}{d_{12}}$, $i = 1, 2$.

We first show that during the above updating process (7.4), the synchronization error converges to a constant determined by the difference between the delays d_{12} and d_{21}.

7.2.1. Theorem. *As t goes to infinity, the difference $x_1(t) - x_2(t)$ between the two clocks converges to $\frac{1}{2}(d_{12} - d_{21})$.*

Proof: Let $e(k) \triangleq x_1(k) - x_2(k)$ for $k \geqslant 0$. Then from (7.4), one has

$$\begin{aligned} e(4k+1) &= \frac{1}{2}e(4k) + \frac{1}{4}(d_{12} - d_{21}) \\ e(4k+2) &= e(4k+1) \\ e(4k+3) &= \frac{1}{2}e(4k+2) + \frac{1}{4}(d_{12} - d_{21}) \\ e(4(k+1)) &= e(4k+3). \end{aligned} \tag{7.5}$$

Substituting the first three equations of (7.5) into the last equation of (7.5), we obtain

$$\begin{aligned} e(4(k+1)) &= (\frac{1}{2})^2 e(4k) + \frac{3}{8}(d_{12} - d_{21}) \\ &= (\frac{1}{2})^{2(k+1)} e(0) + \frac{3}{8}(d_{12} - d_{21}) \sum_{i=0}^{k+1} \frac{1}{4^i}. \end{aligned}$$

Since the geometric series $\sum_{i=0}^{\infty} \frac{1}{4^i}$ converges, we know

$$\lim_{k \to \infty} e(4(k+1)) = \frac{3(d_{12} - d_{21})}{8} \sum_{i=0}^{\infty} \frac{1}{4^i} = \frac{d_{12} - d_{21}}{2}. \tag{7.6}$$

Combining equation (7.6) with (7.5), one can check that

$$\lim_{k\to\infty} e(4k+i) = \frac{1}{2}(d_{12} - d_{21}),\ 1 \leqslant i \leqslant 4. \tag{7.7}$$

From (7.7), we know that for any $\epsilon > 0$, there exists a positive integer M, such that for any $N > M$, $|e(4N+i) - \frac{1}{2}(d_{12} - d_{21})| < \epsilon$, $1 \leqslant i \leqslant 4$. Hence, for any $k > 4(M+1)$, it always holds that $|e(k) - \frac{1}{2}(d_{12} - d_{21})| < \epsilon$, which is equivalent to

$$\lim_{k\to\infty} e(k) = \frac{1}{2}(d_{12} - d_{21}). \tag{7.8}$$

This completes the proof. □

Note that when applying the Network Time Protocol (NTP) [67], it is assumed that most of the time delays are symmetric between a pair of distinct nodes in a network, namely $d_{ij} = d_{ji}$ for $i \neq j$. In fact, in view of Theorem 7.2.1, when $d_{12} = d_{21}$, the two clocks can indeed get synchronized precisely.

7.2.1. Corollary. *When $d_{12} = d_{21}$, the synchronization error $x_1(t) - x_2(t)$ between the two clocks goes to zero asymptotically.*

Now consider the general case when a_2 is different from 1. We first interpret Theorem 7.2.1 in a different way motivated by the approach proposed in [88]. Note that the models of the two clocks with the same skew are

$$x_1(t) = t + b_1,\ x_2(t) = t + b_2.$$

Since the two clocks are with the same skew, to get them synchronized can be regarded as to synchronize the two clocks with respect to a virtual clock

$$x(t) = t + b$$

with b undetermined. Suppose that each clock has an estimate of the virtual clock

$$\hat{x}_1(t) = t + b_1 + o_1(t),\ \hat{x}_2(t) = t + b_2 + o_2(t).$$

Thus the update of the displays of the two clocks in equation (7.4) is equivalent to the update of $o_i(t)$ as follows

$$\begin{cases} o_1(4k+1) = o_1(4k) \\ o_2(4k+1) = o_2(4k) + \frac{1}{2}((\hat{x}_1(4k) + \bar{D}) - (\hat{x}_2(4k) + d_{12})) \end{cases}$$
$$\begin{cases} o_1(4k+2) = o_1(4k+1) \\ o_2(4k+2) = o_2(4k+1) \end{cases}$$
$$\begin{cases} o_1(4k+3) = o_1(4k+2) + \frac{1}{2}((\hat{x}_2(4k+2) + \bar{D}) - (\hat{x}_1(4k+2) + d_{21})) \\ o_2(4k+3) = o_2(4k+2) \end{cases}$$
$$\begin{cases} o_1(4k+4) = o_1(4k+3) \\ o_2(4k+4) = o_2(4k+3), \end{cases}$$

where we use the notation $o_i(k)$ instead of $o_i(t_k)$, $o_1(0) = o_2(0) = 0$, and $o_i(t) = o_i(k)$ for $t \in [t_k, t_{k+1})$. In fact, during the update process, the transmitted time-stamped messages are $\hat{x}_1(t_k)$ and $\hat{x}_2(t_k)$. Then Theorem 7.2.1 says that the difference between the estimates $\hat{x}_1(t) - \hat{x}_2(t) = b_1 + o_1(t) - (b_2 + o_2(t))$ converges to $\frac{1}{2}(d_{12} - d_{21})$ as t goes to infinity.

When the skews of the two clocks are different, consider the models

$$x_1(t) = t + b_1, \; x_2(t) = a_2 t + b_2,$$

where a_2 is close to 1. Since the skew a_2 of clock 2 can be estimated through message passing as shown in (7.2), a transformation of the model of clock 2 leads to the same-skew case

$$\tilde{x}_2(t) = \frac{1}{a_2} x_2(t) = t + \frac{1}{a_2} b_2.$$

Let the estimates of a virtual clock be

$$\hat{x}_1(t) = t + b_1 + o_1(t), \; \hat{x}_2(t) = t + \frac{b_2}{a_2} + o_2(t).$$

From Theorem 7.2.1, one has that $\hat{x}_1(t) - \hat{x}_2(t) = b_1 + o_1(t) - (\frac{b_2}{a_2} + o_2(t))$ converges to $\frac{1}{2}(d_{12} - d_{21})$ as t goes to infinity. In other words, the result stated in Theorem 7.2.1 applies also to the general case when $a_2 \neq 1$.

In the next section, we will study how the main idea of compensation with nominal delays can be applied to larger networks by utilizing the message passing mechanism just described.

7.3 Synchronizing clocks in networks

Now we consider a network of N clocks that are described by (7.1) with $i = 1, \ldots, N$, $a_1 = 1$, and a_i close to 1 for $i = 2, \ldots, N$. Since the skews $a_i, \; i = 2, \ldots, N$, of the clocks can be estimated through message passing, similar to the discussion at the end of Section 7.2, a transformation will lead to the same-skew case

$$\tilde{x}_i(t) = \frac{1}{a_i} x_i(t) = t + \frac{1}{a_i} b_i, \quad i = 2, \ldots, N.$$

Hence, in what follows, we will only consider the case when the skews of the clocks are the same, namely $a_i = 1$ for all i.

Since among the networks with the same number of nodes, the network with a directed ring topology can lead to the greatest difference in the delays of d_{ij} and d_{ji} for a given pair of adjacent nodes i and j, we first study synchronizing clocks in networks with directed ring topologies.

7.3.1 Synchronizing clocks in directed ring networks

A. Synchronizing three clocks in a directed ring network

We fist consider a ring network of three nodes 1, 2 and 3 and three directed edges $(1,2)$, $(2,3)$ and $(3,1)$. Similar to the message passing process for the 2-clock case discussed in the previous section, we illustrate the message passing process among the three clocks in Fig. 7.2, where d_{12}, d_{23}, d_{31} and w_i, $i = 1, 2, 3$, are the time delays and idling times respectively.

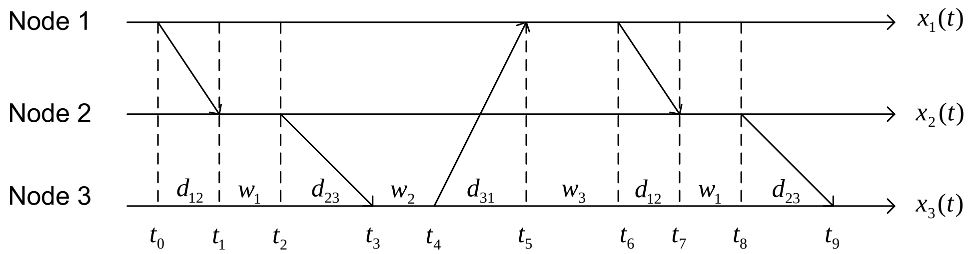


Figure 7.2: *Message exchanges among three clocks with directed connections.*

Although the delays d_{12}, d_{23} and d_{31} cannot be determined from the time-stamped messages, the round-trip delay $D = d_{12} + d_{23} + d_{31}$ can be determined precisely by

$$D = x_1(5) - x_3(4) + x_3(3) - x_2(2) + x_2(1) - x_1(0).$$

We take $\bar{D} = \frac{D}{3}$ as the nominal delay for the three clocks when they update their displays. To be more specific, we take time t_1, when node 2 receives a message from node 1, as an example. At t_1 clock 2 updates its display to the average of its current display and the current estimate of clock 1's display $x_1(0) + \bar{D}$. And w_1 time units later, clock 2 sends the message $x_2(2)$ to clock 3, which in turn updates its display following the same averaging rule. This procedure repeats periodically. As one can see from Fig. 7.2, every link is used exactly once in each period from t_{6k} to $t_{6(k+1)}$ for $k \geqslant 0$.

Now we write down the system equations. Define

$$x(k) = [x_1(k), x_2(k), x_3(k)]^T, \quad v = [d_{12}, d_{23}, d_{31}]^T.$$

Then for $k \geqslant 0$,

$$\begin{aligned}
\begin{bmatrix} x_1(6k+1) \\ x_2(6k+1) \\ x_3(6k+1) \end{bmatrix} &= \begin{bmatrix} x_1(6k) + d_{12} \\ \frac{1}{2}\Big((x_1(6k) + \bar{D}) + (x_2(6k) + d_{12})\Big) \\ x_3(6k) + d_{12} \end{bmatrix} \\
&= \begin{bmatrix} 1 & 0 & 0 \\ \frac{1}{2} & \frac{1}{2} & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x_1(6k) \\ x_2(6k) \\ x_3(6k) \end{bmatrix} + \begin{bmatrix} 1 & 0 & 0 \\ \frac{2}{3} & \frac{1}{6} & \frac{1}{6} \\ 1 & 0 & 0 \end{bmatrix} \begin{bmatrix} d_{12} \\ d_{23} \\ d_{31} \end{bmatrix}.
\end{aligned}$$

Through a similar procedure, one can obtain

$$x(6k+i) = A_i x(6k+i-1) + B_i v, \quad 1 \leqslant i \leqslant 6, \tag{7.9}$$

where

$$\begin{aligned}
A_1 &= \begin{bmatrix} 1 & 0 & 0 \\ \frac{1}{2} & \frac{1}{2} & 0 \\ 0 & 0 & 1 \end{bmatrix}, A_3 = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & \frac{1}{2} & \frac{1}{2} \end{bmatrix}, A_5 = \begin{bmatrix} \frac{1}{2} & 0 & \frac{1}{2} \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}, \\
B_1 &= \begin{bmatrix} 1 & 0 & 0 \\ \frac{2}{3} & \frac{1}{6} & \frac{1}{6} \\ 1 & 0 & 0 \end{bmatrix}, B_3 = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 1 & 0 \\ \frac{1}{6} & \frac{2}{3} & \frac{1}{6} \end{bmatrix}, B_5 = \begin{bmatrix} \frac{1}{6} & \frac{1}{6} & \frac{2}{3} \\ 0 & 0 & 1 \\ 0 & 0 & 1 \end{bmatrix}, \\
A_2 &= A_4 = A_6 = I_3, \; B_{2j} = l_j \begin{bmatrix} \mathbf{1}_3 & O_{3\times 2} \end{bmatrix}, \; j = 1, 2, 3.
\end{aligned}$$

Here, $l_j = \frac{w_j}{d_{12}}$. We can further obtain the following system equation in an iterative form

$$x(6(k+1)) = A_6 A_5 \cdots A_1 x(6k) + \sum_{i=1}^{6} A_6 \cdots A_{i+1} B_i v.$$

Define $A \triangleq A_6 A_5 \cdots A_1 = A_5 A_3 A_1$ and $B \triangleq \sum_{i=1}^{6} A_6 \cdots A_{i+1} B_i$, then we have

$$x(6(k+1)) = A^{k+1} x(0) + \sum_{i=0}^{k} A^i B v, \quad k \geqslant 0. \tag{7.10}$$

We first prove the following convergence result.

7.3.1. Proposition. *As k goes to infinity, $x_i(6(k+1)) - x_j(6(k+1))$ converge to some constants for $i, j = 1, 2, 3,\ i \neq j$.*

For an arbitrary vector $x = [x_1, x_2, \ldots, x_N]^T$, define $\bar{x} = \max_{1 \leqslant i \leqslant N} x_i$ and $\underline{x} = \min_{1 \leqslant i \leqslant N} x_i$. For any matrix $Q = [\alpha_1\ \alpha_2\ \cdots\ \alpha_m]^T \in \mathbb{R}^{m \times m}$, where $\alpha_i \in \mathbb{R}^m,\ i = 1, \ldots, m$, are column vectors, we use

$$\sigma_{ij}(Q) \triangleq \alpha_i^T - \alpha_j^T$$

to denote the difference between the ith and jth rows of Q, where $1 \leqslant i, j \leqslant m,\ i \neq j$. The following result will be useful in the proof for Proposition 7.3.1.

7.3.1. Lemma. *Let $P \in \mathbb{R}^{m \times m}$ be a scrambling matrix. Then $\sigma_{ij}(\sum_{l=0}^{k} P^l)$ converge to some constant row vectors as k goes to infinity for $1 \leqslant i, j \leqslant m,\ i \neq j$.*

Proof: Since the convergence to be proved is meant in the element-wise sense, without loss of generality, we only need to prove the convergence of the first element of $\sigma_{12}(\sum_{l=0}^{k} P^l)$.

Let $(\sigma_{12}(P^i))_1$ be the first element of $\sigma_{12}(P^i)$ for $i \geqslant 0$ and let $(\sigma_{12}(\sum_{i=0}^{k} P^i))_1$ be the first element of $\sigma_{12}(\sum_{i=0}^{k} P^i)$. It is easy to check that the operator $\sigma_{12}(\cdot)$ has the property that

$$(\sigma_{12}(\sum_{i=0}^{k} P^i))_1 = \sum_{i=0}^{k} (\sigma_{12}(P^i))_1.$$

Thus it suffices to prove that $\sum_{i=0}^{\infty} (\sigma_{12}(P^i))_1$ converges. Towards this end, let $u_i = (\sigma_{12}(P^i))_1$ and

$$P^i = \begin{bmatrix} \eta_1^{(i)} & \eta_2^{(i)} & \cdots & \eta_m^{(i)} \end{bmatrix},$$

where $\eta_1^{(i)}, \ldots, \eta_m^{(i)}$ are m column vectors of the matrix P^i. Denote $\overline{\eta_1^{(1)}} = M \geqslant 0$, then one can prove by induction using Lemma 3.3.1(a) that

$$\overline{\eta_1^{(i)}} \leqslant \tau(P)^{i-1} M, \quad i \geqslant 1,$$

where $\tau(P)$ is the coefficient of ergodicity of P.

Let $s_0 = 1$ and $s_i = \tau(P)^{i-1} M,\ i \geqslant 1$, then in view of the definitions of $\sigma_{12}(\cdot)$, one has

$$|u_i| = |(\sigma_{12}(P^i))_1| \leqslant \overline{\eta_1^{(i)}} \leqslant s_i$$

for $i \geqslant 1$. It is obvious that $u_0 = 1$ because $P^0 = I_m$. Then we know $|u_i| \leqslant s_i$ for all $i \geqslant 0$. Since P is a scrambling matrix, $0 \leqslant \tau(P) < 1$, which implies the convergence of the series

$$\sum_{i=0}^{\infty} s_i = 1 + \sum_{i=1}^{\infty} \tau(P)^{i-1} M = 1 + \frac{M}{1 - \tau(P)}.$$

Hence, $\sum_{i=0}^{\infty} |u_i|$ converges, and so does $\sum_{i=0}^{\infty} u_i$. This completes the proof. □

Now we are ready to prove Proposition 7.3.1.
Proof of Proposition 7.3.1: Since $A_1,\ A_3$, and A_5 are all stochastic matrices and the class of all stochastic matrices with the same dimension is closed under multiplication, we know $A = A_5A_3A_1$ is also a stochastic matrix. In addition, because of the special structures of these matrices, one can check that A is scrambling and irreducible [44]. Then we know that $\lim_{k\to\infty} A^k = \lim_{k\to\infty}(A_5A_3A_1)^k = \mathbf{1}_3\zeta^T$ [44], where ζ is some constant column vector. Hence, one immediately gets

$$\lim_{k\to\infty} \sigma_{ij}(A^k) = 0, \quad 1 \leqslant i, j \leqslant 3,\ i \neq j.$$

In view of (7.10), one has

$$x_i(6(k+1)) - x_j(6(k+1)) = \sigma_{ij}(A^{k+1})x(0) + \sigma_{ij}(\sum_{l=0}^{k} A^l)Bv.$$

As $k \to \infty$, it follows from Lemma 7.3.1 that $x_i(6(k+1)) - x_j(6(k+1))$ converge to some constants. □

If we take t_2 or t_4 in Fig. 7.2 as the starting time of the system evolution, following similar arguments as shown above, one can get that $x_i(6k+2) - x_j(6k+2)$ and $x_i(6k+4) - x_j(6k+4)$ both converge to some constants for $1 \leqslant i, j \leqslant 3,\ i \neq j$, as $k \to \infty$. Since

$$x_i(6k+r) - x_j(6k+r) = x_i(6k+r-1) - x_j(6k+r-1),$$

hold for $r = 2, 4, 6$, one can get the following conclusion.

7.3.2. Proposition. *As k goes to infinity, $x_i(6k+r) - x_j(6k+r)$ converge to some constants for all $r = 1, \ldots, 6$, and $i, j = 1, 2, 3,\ i \neq j$.*

From Proposition 7.3.2, we know that we can define

$$e_{ij}(6k+r) \triangleq x_i(6k+r) - x_j(6k+r),$$
$$e(6k+r) \triangleq [e_{12}(6k+r), e_{23}(6k+r)]^T,$$

and the constants

$$e_{ij}^r \triangleq \lim_{k\to\infty} e_{ij}(6k+r),\ \ e^r \triangleq [e_{12}^r, e_{23}^r]^T,$$

where $i, j = 1, 2, 3,\ i \neq j$, and $r = 1, \ldots, 6$. From the system equations (7.9), one can get a set of equations

$$e(6k+i) = \tilde{A}_i e(6k+i-1) + \tilde{B}_i v,\ 1 \leqslant i \leqslant 6, \tag{7.11}$$

where

$$\tilde{A}_1 = \begin{bmatrix} \frac{1}{2} & 0 \\ \frac{1}{2} & 1 \end{bmatrix},\ \tilde{A}_3 = \begin{bmatrix} 1 & 0 \\ 0 & \frac{1}{2} \end{bmatrix},\ \tilde{A}_5 = \begin{bmatrix} \frac{1}{2} & -\frac{1}{2} \\ 0 & 1 \end{bmatrix},$$
$$\tilde{B}_1 = \begin{bmatrix} \frac{1}{3} & -\frac{1}{6} & -\frac{1}{6} \\ -\frac{1}{3} & \frac{1}{6} & \frac{1}{6} \end{bmatrix},\ \tilde{B}_3 = \begin{bmatrix} 0 & 0 & 0 \\ -\frac{1}{6} & \frac{1}{3} & -\frac{1}{6} \end{bmatrix},\ \tilde{B}_5 = \begin{bmatrix} \frac{1}{6} & \frac{1}{6} & -\frac{1}{3} \\ 0 & 0 & 0 \end{bmatrix},$$
$$\tilde{A}_2 = \tilde{A}_4 = \tilde{A}_6 = I_2,\ \tilde{B}_2 = \tilde{B}_4 = \tilde{B}_6 = O_{2\times 3}.$$

By iteration, one has

$$\begin{aligned} e(6(k+1)) &= \tilde{A}_6\tilde{A}_5\cdots\tilde{A}_1 e(6k) + \sum_{i=1}^{6} \tilde{A}_6\cdots\tilde{A}_{i+1}\tilde{B}_i v \\ &= \tilde{A}^{k+1}e(0) + \sum_{i=0}^{k} \tilde{A}^i\tilde{B}v, \quad k \geqslant 0, \end{aligned}$$

where $\tilde{A} \triangleq \tilde{A}_6\tilde{A}_5\cdots\tilde{A}_1 = \tilde{A}_5\tilde{A}_3\tilde{A}_1$ and $\tilde{B} = \sum_{i=1}^{6}\tilde{A}_6\cdots\tilde{A}_{i+1}\tilde{B}_i$. Taking k to infinity, one has

$$\lim_{k\to\infty} e(6(k+1)) = \lim_{k\to\infty} \tilde{A}^{k+1}e(0) + \lim_{k\to\infty}\sum_{i=0}^{k}\tilde{A}^i\tilde{B}v,\ k \geqslant 0.$$

Since the limit $\lim_{k\to\infty} e(6(k+1))$ exists for any initial condition and any time delays from Proposition 7.3.2, it must be true that both $\lim_{k\to\infty}\tilde{A}^{k+1}$ and $\lim_{k\to\infty}\sum_{i=0}^{k}\tilde{A}^i$ converge, from which we conclude that $\rho(\tilde{A}) < 1$, namely, the spectral radius of $\tilde{A}$ is strictly less than 1.

In view of the fact that $e^{r+1} = e^r,\ r = 1, 3, 5$, we define

$$e \triangleq [(e^1)^T, (e^3)^T, (e^5)^T]^T.$$

Then we get the equation of the asymptotic synchronization errors between clocks by taking k on both sides of (7.11) to infinity:

$$e = \bar{A}e + \bar{B}v,$$

where

$$\bar{A} = \begin{bmatrix} O & O & \tilde{A}_1 \\ \tilde{A}_3 & O & O \\ O & \tilde{A}_5 & O \end{bmatrix},\ \bar{B} = \begin{bmatrix} \tilde{B}_1 \\ \tilde{B}_3 \\ \tilde{B}_5 \end{bmatrix}.$$

If the matrix $I - \bar{A}$ is invertible, the error e can be calculated as $e = (I - \bar{A})^{-1}\bar{B}v$.

7.3.2. Lemma. *The matrix $I - \bar{A}$ is invertible.*

Proof: We prove this Lemma by showing that $(I - \bar{A})y = 0$ has a unique solution $y = 0$. From

$$(I - \bar{A})y = \begin{bmatrix} I & O & -\tilde{A}_1 \\ -\tilde{A}_3 & I & O \\ O & -\tilde{A}_5 & I \end{bmatrix} \begin{bmatrix} y_1 \\ y_2 \\ y_3 \end{bmatrix} = 0,$$

one has

$$y_1 = \tilde{A}_1 y_3,\ y_2 = \tilde{A}_3 y_1,\ y_3 = \tilde{A}_5 y_2, \tag{7.12}$$

Substituting the first two equations of (7.12) into the last one of (7.12), we obtain

$$y_3 = \tilde{A}_5 \tilde{A}_3 \tilde{A}_1 y_3 = \tilde{A} y_3.$$

Since $\rho(\tilde{A}) < 1$, $y_3 = \tilde{A} y_3$ has a unique solution $y_3 = 0$. Substituting $y_3 = 0$ into the equations (7.12), one has $y = 0$. □

Thus, by calculating $e = (I - \bar{A})^{-1} \bar{B} v$, one has

$$e_{12}^r = d_{12} - \bar{D},\ e_{23}^r = d_{23} - \bar{D},\ r = 1, \ldots, 6.$$

Hence, we have proved the following.

7.3.1. THEOREM. *As time goes to infinity, the synchronization errors between clocks in the three-clock directed ring network converge and*

$$\lim_{t \to \infty} (x_i(t) - x_{[i]}(t)) = d_{i,[i]} - \bar{D}, \quad i = 1, 2, 3,$$

where $[i] = i + 1$ *if* $i = 1, 2$ *and* $[i] = 1$ *if* $i = 3$.

The following result is a direct consequence of Theorem 7.3.1.

7.3.1. COROLLARY. *For the three clocks in the directed ring network, if the delays are all equal, namely* $d_{12} = d_{23} = d_{31}$*, the clocks can get synchronized asymptotically.*

In the next subsection, we extend the results that we have obtained for the three-clock directed ring network to general directed ring networks with $n \geqslant 3$ nodes.

B. Synchronizing more clocks in a directed ring network

Now we consider a directed ring network of $N \geqslant 3$ nodes. The message passing procedure in the network with unidirectional communications is illustrated in Fig. 7.3, where $d_{i,[i]}$ and w_i, $i = 1, \ldots, n$, are time delays and idling times respectively. Here, $[i]$ is defined to be $i + 1$ when $i = 1, \ldots, N - 1$ and 1 when $i = N$.

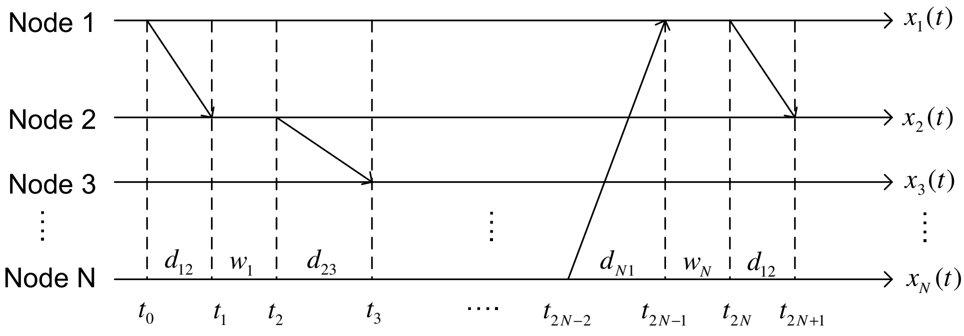


Figure 7.3: *Message exchanges among $N \geqslant 3$ clocks with directed connections.*

Although the time delays $d_{i,[i]}$, $i = 1, \ldots, N$, between clocks cannot be determined precisely no matter how many time-stamped messages are exchanged, the round-trip delay $D = \sum_{i=1}^{N} d_{i,[i]}$ can be calculated after sufficiently many messages are delivered

$$D = \sum_{i=0}^{N-1} \Big(x_{[i+1]}(2i+1) - x_{i+1}(2i) \Big).$$

Similar to the three-clock case in Subsection 7.3.1A, we use $\bar{D} = \frac{D}{N}$ as the nominal delay for all the clocks when they update their displays.

Define

$$x(k) = [x_1(k), x_2(k), \ldots, x_N(k)]^T, \; v = [d_{12}, d_{23}, \ldots, d_{N1}]^T.$$

Then we have the system equations in state space

$$x(2Nk + i) = A_i x(2Nk + i - 1) + B_i v, \tag{7.13}$$

for $1 \leqslant i \leqslant 2N$ and $k \geqslant 0$, where

$$A_1 = \begin{bmatrix} 1 & 0 & 0 & \cdots & 0 \\ \frac{1}{2} & \frac{1}{2} & 0 & \cdots & 0 \\ 0 & 0 & 1 & \ddots & 0 \\ \vdots & \vdots & \ddots & \ddots & \vdots \\ 0 & 0 & 0 & \cdots & 1 \end{bmatrix}, \dots, \; A_{2N-1} = \begin{bmatrix} \frac{1}{2} & 0 & 0 & \cdots & \frac{1}{2} \\ 0 & 1 & 0 & \cdots & 0 \\ 0 & 0 & 1 & \ddots & 0 \\ \vdots & \vdots & \ddots & \ddots & \vdots \\ 0 & 0 & 0 & \cdots & 1 \end{bmatrix},$$

$$B_1 = \begin{bmatrix} 1 & 0 & 0 & \cdots & 0 \\ \frac{N+1}{2N} & \frac{1}{2N} & \frac{1}{2N} & \cdots & \frac{1}{2N} \\ 1 & 0 & 0 & \ddots & 0 \\ \vdots & \vdots & \ddots & \ddots & \vdots \\ 1 & 0 & 0 & \cdots & 0 \end{bmatrix}, \dots, \; B_{2N-1} = \begin{bmatrix} \frac{1}{2N} & \frac{1}{2N} & \frac{1}{2N} & \cdots & \frac{N+1}{2N} \\ 0 & 0 & 0 & \cdots & 1 \\ 0 & 0 & 0 & \ddots & 1 \\ \vdots & \vdots & \ddots & \ddots & \vdots \\ 0 & 0 & 0 & \cdots & 1 \end{bmatrix},$$

$$A_2 = A_4 = \cdots = A_{2N} = I_N, \; B_{2j} = l_j \begin{bmatrix} \mathbf{1}_N & O_{N \times (N-1)} \end{bmatrix}, \; j = 1, \dots, N,$$

and $l_j = \frac{w_j}{d_{12}}$.

We can write down the iterative equations

$$\begin{aligned} x(2N(k+1)) &= A_{2N} A_{2N-1} \cdots A_1 x(2Nk) + (\sum_{i=1}^{2N} A_{2N} \cdots A_{i+1} B_i) v \\ &= A^{k+1} x(0) + \sum_{i=0}^{k} A^i B v. \end{aligned}$$

where $A = A_{2N} A_{2N-1} \cdots A_1$ and $B = \sum_{i=1}^{2N} A_{2N} \cdots A_{i+1} B_i$. This equation is in the same form as Eq. (7.10). Then using similar arguments to that in Subsection 7.3.1A, one can prove the following result.

7.3.2. Theorem. *As time goes to infinity, the synchronization errors between clocks in the N-clock ring network, $N \geqslant 3$, converge and*

$$\lim_{t \to \infty} (x_i(t) - x_{[i]}(t)) = d_{i,[i]} - \bar{D}, \quad i = 1, \dots, N.$$

Since undirected graphs can be viewed as a special class of directed graphs, the 2-clock synchronization discussed in Section 7.2 can be viewed as a special case of the N-clock synchronization in a directed ring network when $N = 2$. In view of this, Theorem 7.2.1 is consistent with Theorem 7.3.2.

In the next subsection, we discuss how to synchronize clocks in connected undirected networks.

7.3.2 Synchronizing clocks in connected undirected networks

A. Synchronizing three clocks in a connected undirected network

We fist consider a network of three nodes with undirected edges $(1,2)$, $(2,3)$ and $(1,3)$. Similar to the message passing process for the 2-clock case discussed before, we illustrate the message passing process among the three clocks in Fig. 7.4.

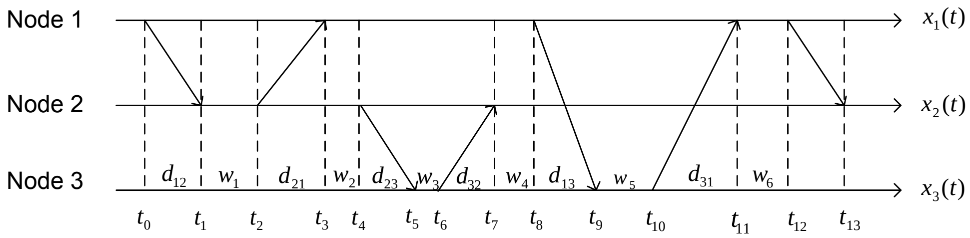


Figure 7.4: *Message exchanges among three clocks with undirected connections.*

Although the delays d_{ij}, $1 \leqslant i,j \leqslant 3$, cannot be determined from the time-stamped messages, the round-trip delay between each pair of connected clocks can be calculated precisely. For example, the round-trip delay D_{12} between clocks 1 and 2 is

$$D_{12} = d_{12} + d_{21} = x_1(3) - x_2(2) + x_2(1) - x_1(0).$$

We take $\bar{D}_{ij} = \frac{D_{ij}}{2}$ as the nominal delay for a pair of adjacent clocks i and j when they update their displays, where $D_{ij} = d_{ij} + d_{ji}$ is the round-trip delay between clocks i and j. As before the clocks update following the same average rule and this procedure repeats periodically. It can be seen from Fig. 7.4 that, in each update period from t_{12k} to $t_{12(k+1)}$ for $k \geqslant 0$, a pair of adjacent nodes exchange messages exactly once.

Define $x(k) = [x_1(k), x_2(k), x_3(k)]^T$ and $v = [d_{12}, d_{21}, d_{23}, d_{32}, d_{13}, d_{31}]^T$. Then we obtain the system equations

$$x(12k+i) = A_i x(12k+i-1) + B_i v, \tag{7.14}$$

for $1 \leqslant i \leqslant 12$ and $k \geqslant 0$, where

$$A_1 = \begin{bmatrix} 1 & 0 & 0 \\ \frac{1}{2} & \frac{1}{2} & 0 \\ 0 & 0 & 1 \end{bmatrix},\ A_3 = \begin{bmatrix} \frac{1}{2} & \frac{1}{2} & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix},\ A_5 = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & \frac{1}{2} & \frac{1}{2} \end{bmatrix},$$

$$A_7 = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \frac{1}{2} & \frac{1}{2} \\ 0 & 0 & 1 \end{bmatrix},\ A_9 = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ \frac{1}{2} & 0 & \frac{1}{2} \end{bmatrix},\ A_{11} = \begin{bmatrix} \frac{1}{2} & 0 & \frac{1}{2} \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix},$$

$$B_1 = \begin{bmatrix} \begin{matrix} 1 & 0 \\ \frac{3}{4} & \frac{1}{4} \\ 1 & 0 \end{matrix} & O_{3\times 4} \end{bmatrix},\ B_3 = \begin{bmatrix} \begin{matrix} \frac{1}{4} & \frac{3}{4} \\ 0 & 1 \\ 0 & 1 \end{matrix} & O_{3\times 4} \end{bmatrix},\ B_5 = \begin{bmatrix} 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & \frac{3}{4} & \frac{1}{4} & 0 & 0 \end{bmatrix},$$

$$B_7 = \begin{bmatrix} 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & \frac{1}{4} & \frac{3}{4} & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \end{bmatrix},\ B_9 = \begin{bmatrix} O_{3\times 4} & \begin{matrix} 1 & 0 \\ 1 & 0 \\ \frac{3}{4} & \frac{1}{4} \end{matrix} \end{bmatrix},\ B_{11} = \begin{bmatrix} O_{3\times 4} & \begin{matrix} \frac{1}{4} & \frac{3}{4} \\ 0 & 1 \\ 0 & 1 \end{matrix} \end{bmatrix},$$

$$A_2 = \cdots = A_{12} = I_3,\ B_{2j} = l_j \begin{bmatrix} \mathbf{1}_3 & O_{3\times 5} \end{bmatrix},\ 1 \leqslant j \leqslant 6.$$

Here $l_j = \frac{w_j}{d_{12}}$. By iteration, we have

$$x(12(k+1)) = A^{k+1}x(0) + \sum_{i=0}^{k} A^i B v, \quad k \geqslant 0,$$

where $A = A_{12}A_{11}\cdots A_1$ and $B = \sum_{i=1}^{12} A_{12}\cdots A_{i+1}B_i$. Following similar arguments to that in Subsection 7.3.1A, one can prove the following result.

7.3.3. Proposition. *As k goes to infinity, $x_i(12k+r) - x_j(12k+r)$ converge to some constants for all $r = 1, \ldots, 12$, and $i, j = 1, 2, 3,\ i \neq j$.*

Define

$$e_{ij}(12k+r) \triangleq x_i(12k+r) - x_j(12k+r),$$
$$e(12k+r) \triangleq [e_{12}(12k+r), e_{23}(12k+r)]^T,$$

and the constants

$$e_{ij}^r \triangleq \lim_{k\to\infty} e_{ij}(12k+r),\ e^r \triangleq [e_{12}^r, e_{23}^r]^T,$$

where $i, j = 1, 2, 3,\ i \neq j$, and $r = 1, \ldots, 12$. From the system equations (7.14), one gets a set of equations

$$e(12k+i) = \tilde{A}_i e(12k+i-1) + \tilde{B}_i v,\ 1 \leqslant i \leqslant 12, \tag{7.15}$$

where

$$
\begin{aligned}
&\tilde{A}_1 = \begin{bmatrix} \frac{1}{2} & 0 \\ \frac{1}{2} & 1 \end{bmatrix}, \ \tilde{A}_3 = \begin{bmatrix} \frac{1}{2} & 0 \\ 0 & 1 \end{bmatrix}, \ \tilde{A}_5 = \begin{bmatrix} 1 & 0 \\ 0 & \frac{1}{2} \end{bmatrix}, \\
&\tilde{A}_7 = \begin{bmatrix} 1 & \frac{1}{2} \\ 0 & \frac{1}{2} \end{bmatrix}, \ \tilde{A}_9 = \begin{bmatrix} 1 & 0 \\ -\frac{1}{2} & \frac{1}{2} \end{bmatrix}, \ \tilde{A}_{11} = \begin{bmatrix} \frac{1}{2} & -\frac{1}{2} \\ 0 & 1 \end{bmatrix}, \\
&\tilde{B}_1 = \frac{1}{4}\begin{bmatrix} \begin{matrix} 1 & -1 \\ -1 & 1 \end{matrix} & O_{2\times 4} \end{bmatrix}, \ \tilde{B}_3 = \frac{1}{4}\begin{bmatrix} \begin{matrix} 1 & -1 \\ 0 & 0 \end{matrix} & O_{2\times 4} \end{bmatrix}, \\
&\tilde{B}_5 = \frac{1}{4}\begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & -1 & 0 & 0 \end{bmatrix}, \ \tilde{B}_7 = \frac{1}{4}\begin{bmatrix} 0 & 0 & -1 & 1 & 0 & 0 \\ 0 & 0 & 1 & -1 & 0 & 0 \end{bmatrix}, \\
&\tilde{B}_9 = \frac{1}{4}\begin{bmatrix} O_{2\times 4} & \begin{matrix} 0 & 0 \\ 1 & -1 \end{matrix} \end{bmatrix}, \ \tilde{B}_{11} = \frac{1}{4}\begin{bmatrix} O_{2\times 4} & \begin{matrix} 1 & -1 \\ 0 & 0 \end{matrix} \end{bmatrix}, \\
&\tilde{A}_2 = \cdots = \tilde{A}_{12} = I_2, \ \tilde{B}_2 = \cdots = \tilde{B}_{12} = O_{2\times 6}.
\end{aligned}
$$

Since $e^{r+1} = e^r, \ r = 1, 3, \ldots, 11$, we conclude from Proposition 7.3.3 that as $k \to \infty$, the synchronization errors between a pair of distinct nodes approach permanent oscillations among at most 6 values. One can further calculate these values easily.

Let $e \triangleq [(e^1)^T, (e^3)^T, \ldots, (e^{11})^T]^T$. By taking k on both sides of (7.15) to infinity, we can get the equation for the synchronization errors between clocks

$$e = \bar{A}e + \bar{B}v, \tag{7.16}$$

where

$$
\bar{A} = \begin{bmatrix} O & O & \cdots & O & \tilde{A}_1 \\ \tilde{A}_3 & O & \cdots & O & O \\ \vdots & \ddots & \ddots & \vdots & \vdots \\ O & O & \ddots & O & O \\ O & O & \cdots & \tilde{A}_{11} & O \end{bmatrix}, \ \bar{B} = \begin{bmatrix} \tilde{B}_1 \\ \tilde{B}_3 \\ \vdots \\ \tilde{B}_9 \\ \tilde{B}_{11} \end{bmatrix}.
$$

Since $(I - \bar{A})$ is invertible, which can be proved using similar arguments to that in Lemma 7.3.2, the error e can be calculated by $e = (I - \bar{A})^{-1}\bar{B}v$. Thus we have proved the following result.

7.3.3. THEOREM. *As time goes to infinity, the synchronization errors between each pair of distinct clocks in the three-clock connected undirected network will approach permanent oscillations among at most* 6 *values, which are determined by*

$$e = (I - \bar{A})^{-1}\bar{B}v.$$

7.3.1. REMARK. In the three-clock directed ring network, the synchronization errors between clocks converge to some constants; for example, $\lim_{t\to\infty}(x_1(t) - x_2(t)) = d_{12} - \bar{D} = e_{12}^r$ for all $r = 1, \ldots, 6$. However, in the three-clock connected undirected network, the synchronization errors between a pair of distinct clocks may not converge, which in general will oscillate; for example, $\lim_{t\to\infty}(x_1(t) - x_2(t))$ may not exist because $e_{12}^{r_1}$ may not be equal to $e_{12}^{r_2}$ for some $r_1,\ r_2,\ 1 \leqslant r_1, r_2 \leqslant 12$.

Although the synchronization errors between a pair of distinct clocks in general will oscillate, it is easy to see that if

$$e^{r_1} = e^{r_2},\ \forall r_1, r_2 = 1, 3, \ldots, 11, \tag{7.17}$$

then the errors converge to some constant values. Substituting (7.17) into (7.16), one has

$$e_{12}^r = d_{12} - \bar{D}_{12},\ e_{23}^r = d_{23} - \bar{D}_{23},\ e_{12}^r + e_{23}^r = d_{13} - \bar{D}_{13},$$

where $r = 1, 3, \ldots, 11$. Since (7.16) has a unique solution $e = (I - \bar{A})^{-1}\bar{B}v$, we can conclude that if $d_{12} - \bar{D}_{12} + d_{23} - \bar{D}_{23} = d_{13} - \bar{D}_{13}$, namely, $d_{12} + d_{23} + d_{31} = d_{13} + d_{32} + d_{21}$, then $e_{12}^r = d_{12} - \bar{D}_{12},\ e_{23}^r = d_{23} - \bar{D}_{23}$, $r = 1, 3, \ldots, 11$, is indeed the solution to (7.16). We summarize.

7.3.2. COROLLARY. *If $d_{12} + d_{23} + d_{31} = d_{13} + d_{32} + d_{21}$, then as time goes to infinity, the synchronization errors between clocks in the three-clock undirected network converge and*

$$\lim_{t\to\infty}(x_i(t) - x_j(t)) = d_{ij} - \bar{D}_{ij},\ i \neq j.$$

Specifically, if the time delays are symmetric, namely $d_{ij} = d_{ji}$ $i \neq j$, then the three clocks can get synchronized asymptotically.

In the next subsection, we extend the results that we have obtained for the three-clock connected network to general connected networks with bidirectional links.

B. Synchronizing more clocks in a connected undirected network

We consider a connected network consisting of N nodes and m undirected edges. For the ease of describing the message passing process, we assume that the edges have been labeled and in each update period, a pair of connected nodes exchange messages exactly once. The indices of the edges determine the ordering of the pair of nodes that are activated to exchange messages. For the two nodes associated with an edge, the one with the smaller index starts the message exchange process. For the sth edge of the graph, let $s_1 < s_2$ denote the indices of the associated two nodes. Then s_1 always sends a message to s_2 first, and then s_2 replies. Taking the three clocks in Subsection 7.3.2A as an example, we label the edges $(1,2)$, $(2,3)$ and $(1,3)$ by ①, ② and ③, respectively. For the 2nd edge $(2,3)$, node $②_1 = 2$ always sends a message to node $②_2 = 3$ first, and after waiting for some idling time, node 3 sends back a message to node 2. Thus the message passing process is illustrated more in detail in Fig. 7.4.

Define $x(k) = [x_1(k), \dots, x_N(k)]^T$, and $v = [d_{1_1,1_2}, d_{1_2,1_1}, \dots, d_{m_1,m_2}, d_{m_2,m_1}]^T$. Then we can derive the system equations through a similar procedure to that in Subsection 7.3.1A.

$$x(4mk+i) = A_i x(4mk+i-1) + B_i v, \ 1 \leqslant i \leqslant 4m, \ k \geqslant 0, \tag{7.18}$$

where

$$A_2 = \cdots = A_{4m} = I_N, \ B_{2j} = l_j \begin{bmatrix} \mathbf{1}_N & O_{N\times(2m-1)} \end{bmatrix},$$

and $l_j = \frac{w_j}{d_{1_1,1_2}}$, w_j are idling times for $1 \leqslant j \leqslant 2m$, and when $i = 4(s-1)+1$, for $1 \leqslant s \leqslant m$,

$$A_i = diag\{I_{s_1-1}, A_i', I_{N-s_2}\}, \ B_i = \begin{bmatrix} O_{N\times(2s-2)} & B_i' & O_{N\times(2m-2s)} \end{bmatrix},$$

with

$$A_i' = \begin{bmatrix} 1 & 0 & \cdots & 0 \\ 0 & 1 & \ddots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ \frac{1}{2} & 0 & \cdots & \frac{1}{2} \end{bmatrix}, \ B_i' = \begin{bmatrix} \mathbf{1}_{s_1-1} & \mathbf{0}_{s_1-1} \\ 1 & 0 \\ \vdots & \vdots \\ \frac{3}{4} & \frac{1}{4} \\ \mathbf{1}_{N-s_2} & \mathbf{0}_{N-s_2}, \end{bmatrix}.$$

When $i = 4(s-1)+3$, for $1 \leqslant s \leqslant m$,

$$A_i = diag\{I_{s_1-1}, A_i', I_{N-s_2}\}, \ B_i = \begin{bmatrix} O_{N\times(2s-2)} & B_i' & O_{N\times(2m-2s)} \end{bmatrix},$$

with

$$A_i' = \begin{bmatrix} \frac{1}{2} & 0 & \cdots & \frac{1}{2} \\ 0 & 1 & \ddots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & 1 \end{bmatrix}, \ B_i' = \begin{bmatrix} \mathbf{0}_{s_1-1} & \mathbf{1}_{s_1-1} \\ \frac{1}{4} & \frac{3}{4} \\ \vdots & \vdots \\ 0 & 1 \\ \mathbf{0}_{N-s_2} & \mathbf{1}_{N-s_2}, \end{bmatrix}.$$

We can further obtain

$$x(4m(k+1)) = A^{k+1}x(0) + \sum_{i=0}^{k} A^i B v, \quad k \geqslant 0,$$

where $A = A_{4m}A_{4m-1}\cdots A_1$ and $B = \sum_{i=1}^{4m} A_{4m}\cdots A_{i+1}B_i$. Following similar arguments to that in Subsection 7.3.1A, we can conclude that as k goes to infinity,

$x_i(4mk+r) - x_j(4mk+r)$ converge to some constants for all $r = 1, \ldots, 4m$, and $i, j = 1, \ldots, N,\ i \neq j$.

Then define $e_{ij}(4mk+r) \triangleq x_i(4mk+r) - x_j(4mk+r)$, $e(4mk+r) \triangleq [e_{12}(4mk+r), e_{23}(4mk+r), \ldots, e_{N-1,N}(4mk+r)]^T$, and the constants $e^r_{ij} \triangleq \lim_{k\to\infty} e_{ij}(4mk+r)$, $e^r \triangleq [e^r_{12}, e^r_{23}, \ldots, e^r_{N-1,N}]^T$, where $i, j = 1, \ldots, N,\ i \neq j$, and $r = 1, \ldots, 4m$. From the system equations (7.18), one can get a set of equations

$$e(4mk+i) = \tilde{A}_i e(4mk+i-1) + \tilde{B}_i v,\ 1 \leqslant i \leqslant 4m, \tag{7.19}$$

where

$$\tilde{A}_2 = \cdots = \tilde{A}_{4m} = I_{N-1},\ \tilde{B}_2 = \cdots = \tilde{B}_{4m} = O_{(N-1)\times 2m},$$

and when $i = 4(s-1)+1$, for $1 \leqslant s \leqslant m$,

$$\tilde{A}_i = diag\{I_{s_1-1}, \tilde{A}'_i, I_{N-1-s_2}\},\ \tilde{B}_i = \begin{bmatrix} O_{(N-1)\times(2s-2)} & \tilde{B}'_i & O_{(N-1)\times(2m-2s)} \end{bmatrix},$$

with

$$\tilde{A}'_i = \begin{bmatrix} 1 & \cdots & 0 & 0 & 0 \\ \vdots & \ddots & \vdots & \vdots & \vdots \\ 0 & \cdots & 1 & 0 & 0 \\ -\frac{1}{2} & \cdots & -\frac{1}{2} & \frac{1}{2} & 0 \\ \frac{1}{2} & \cdots & \frac{1}{2} & \frac{1}{2} & 1 \end{bmatrix},\ \tilde{B}'_i = \begin{bmatrix} \mathbf{0}_{s_2-2} & \mathbf{0}_{s_2-2} \\ \frac{1}{4} & -\frac{1}{4} \\ -\frac{1}{4} & \frac{1}{4} \\ \mathbf{0}_{N-1-s_2} & \mathbf{0}_{N-1-s_2} \end{bmatrix},$$

when $i = 4(s-1)+3$, for $1 \leqslant s \leqslant m$,

$$\tilde{A}_i = diag\{I_{s_1-2}, \tilde{A}'_i, I_{N-s_2}\},\ \tilde{B}_i = \begin{bmatrix} O_{(N-1)\times(2s-2)} & \tilde{B}'_i & O_{(N-1)\times(2m-2s)} \end{bmatrix},$$

with

$$\tilde{A}'_i = \begin{bmatrix} 1 & \frac{1}{2} & \frac{1}{2} & \cdots & \frac{1}{2} \\ 0 & \frac{1}{2} & -\frac{1}{2} & \cdots & -\frac{1}{2} \\ 0 & 0 & 1 & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \cdots & 1 \end{bmatrix},\ \tilde{B}'_i = \begin{bmatrix} \mathbf{0}_{s_1-2} & \mathbf{0}_{s_1-2} \\ -\frac{1}{4} & \frac{1}{4} \\ \frac{1}{4} & -\frac{1}{4} \\ \mathbf{0}_{N-s_1-1} & \mathbf{0}_{N-s_1-1} \end{bmatrix}.$$

Let

$$e \triangleq [(e^1)^T, (e^3)^T, \ldots, (e^{4m-1})^T]^T.$$

Then the equation of the synchronization errors can be written as

$$e = \bar{A}e + \bar{B}v,$$

where

$$\bar{A} = \begin{bmatrix} O & O & \cdots & O & \tilde{A}_1 \\ \tilde{A}_3 & O & \cdots & O & O \\ \vdots & \ddots & \ddots & \vdots & \vdots \\ O & O & \ddots & O & O \\ O & O & \cdots & \tilde{A}_{4m-1} & O \end{bmatrix}, \ \bar{B} = \begin{bmatrix} \tilde{B}_1 \\ \tilde{B}_3 \\ \vdots \\ \tilde{B}_{4m-3} \\ \tilde{B}_{4m-1} \end{bmatrix}.$$

Since the matrix $I - \bar{A}$ is invertible, the errors can be calculated by $e = (I - \bar{A})^{-1}\bar{B}v$.

7.3.4. THEOREM. *As time goes to infinity, the synchronization errors between each pair of distinct clocks in the N-clock undirected connected network will approach permanent oscillations among at most $2m$ values, which are determined by*

$$e = (I - \bar{A})^{-1}\bar{B}v.$$

Networks with tree topologies are preferred when applying network clock synchronization protocols [34], the following corollary suggests the reason behind it.

7.3.3. COROLLARY. *If the communication graph $\mathbb{G}$ is an undirected tree, the synchronization errors between clocks in the network converge and*

$$\lim_{t\to\infty}(x_i(t) - x_j(t)) = d_{ij} - \bar{D}_{ij}, \ i \neq j,$$

where $(i, j) \in \mathcal{E}$ and $\bar{D}_{ij} = \frac{1}{2}(d_{ij} + d_{ji})$.

In the next section, we discuss how to synchronize clocks in networks with strongly connected directed topologies.

7.3.3 Expansion to strongly connected directed networks

In order to synchronize N clocks in a network with strongly connected directed topology, we may use only some of the edges in the network. To better explain this idea, we need to introduce some more notions.

For a strongly connected graph $\mathbb{G} = (\mathcal{V}, \mathcal{E})$, we can find subgraphs $\mathbb{G}_i = (\mathcal{V}_i, \mathcal{E}_i), \ i = 1, \ldots, p$, of $\mathbb{G}$ such that $\cup_{i=1}^{p}\mathcal{V}_i = \mathcal{V}$ and each $\mathbb{G}_i$ is a directed ring graph. Those edges in $\cup_{i=1}^{p}\mathcal{E}_i$ are to be utilized in the message passing process. We divide each update period of the overall network into p stages. Each stage corresponds to a directed ring subgraph $\mathbb{G}_i$, in which the message passing process is the same as that in Subsection 7.3.1B. Note that $\mathbb{G}_i$ might share common edges and the nodes associated with these edges will carry out message passing more than once in each period. We take the message passing process in Subsection 7.3.2A as an example since connected undirected graphs can always be viewed as strongly connected directed graphs. The graph corresponds to Fig. 7.4 is $\mathbb{G} = (\mathcal{V}, \mathcal{E})$ with $\mathcal{V} = \{1, 2, 3\}$ and $\mathcal{E} = \{(1, 2), (2, 1), (2, 3), (3, 2), (1, 3), (3, 1)\}$. Define $\mathbb{G}_i = (\mathcal{V}_i, \mathcal{E}_i),\ i = 1, 2, 3$, with $\mathcal{V}_1 = \{1, 2\}$, $\mathcal{E}_1 = \{(1, 2), (2, 1)\}$, $\mathcal{V}_2 = \{2, 3\}$, $\mathcal{E}_2 = \{(2, 3), (3, 2)\}$, and $\mathcal{V}_3 = \{1, 3\}$, $\mathcal{E}_3 = \{(1, 3), (3, 1)\}$. It is easy to check that $\cup_{i=1}^{3} \mathcal{V}_i = \mathcal{V}$ and $\mathbb{G}_i$ are directed ring graphs for $i = 1, 2, 3$. Thus each update period can be divided into 3 stages, and each stage corresponds to a subgraph $\mathbb{G}_i$. The message passing process in each stage is the same as that in Subsection 7.3.1B.

One can obtain the following result which is similar to that in the previous section.

7.3.5. Theorem. *As time goes to infinity, the synchronization errors between each pair of distinct clocks in the N-clock strongly connected network will approach permanent oscillations among at most $\sum_{i=1}^{p} |\mathcal{E}_i|$ values.*

The synchronization errors between clocks, which in general will oscillate, are determined by the choices of the subgraphs and the time delays. A proper choice of the subgraphs can lead to the convergence of synchronization errors. One example is that if we only choose the subgraphs $\mathbb{G}_1$, $\mathbb{G}_2$ of $\mathbb{G}$ defined in the previous example for message passing, the synchronization errors will converge to some constants in view of Corollary 7.3.3.

7.4 Illustrative examples

7.4.1. Example. (*Directed ring networks*) We first consider three clocks with the same skew in a directed ring network, whose message passing procedure is shown in Fig. 7.2. The three time delays, not known by the clocks, are $d_{12} = 0.2,\ d_{23} = 0.4$, and $d_{31} = 0.9$. Then the round-trip delay $D = 1.5$ and the nominal delay $\bar{D} = 0.5$. Every clock waits for one time unit after receiving a message before sending its own message, namely $w_i = 1,\ i = 1, 2, 3$. We set the initial time displays of the three clocks to be $[x_1(0), x_2(0), x_3(0)]^T = [10, 40, 20]^T$. The simulation results of the evolution of the displays of the three clocks are shown in Fig. 7.5. One can see that the three clocks do not synchronize, but the asymptotic synchronization error between clocks 1 and 2 is -0.3 and that between clocks 2 and 3 is -0.1, which agrees with our theoretical analysis. If we set all the three time delays to be equal, namely $d_{12} = d_{23} = d_{31} = 0.5$, then from Theorem 7.3.1 it follows that the three clocks are synchronized asymptotically as shown in Fig. 7.6.

Since the time delays are random variables in real distributed networks, we rerun the simulation for the case when the delays take random values in the intervals $d_{12} \in [0.15,\ 0.25]$, $d_{23} \in [0.3,\ 0.5]$, and $d_{31} \in [0.8,\ 1.0]$. The expected round-trip

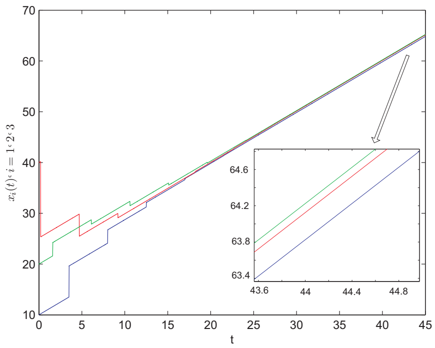


Figure 7.5: *Time displays of three clocks with nonidentical delays in a directed ring network.*

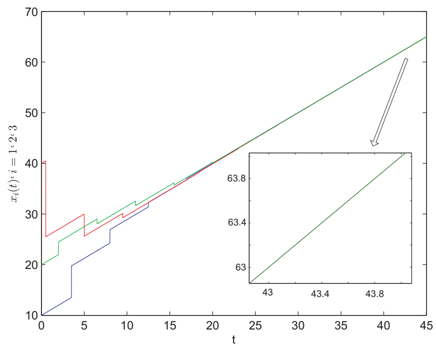


Figure 7.6: *Time displays of three clocks with identical delays in a directed ring network.*

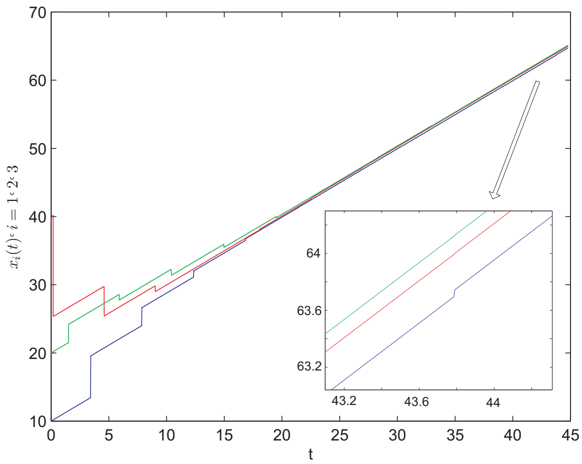


Figure 7.7: *Time displays of three clocks with time-varying delays in a directed ring network.*

delay is still $D = 1.5$, and thus $\bar{D}$ is still 0.5. The simulation results are shown in Fig. 7.7, from which one can tell the clock synchronization errors are bounded in a small range.

7.4.2. Example. (*Connected undirected networks*) We consider three clocks with the same skew in an undirected network, whose message passing procedure is shown

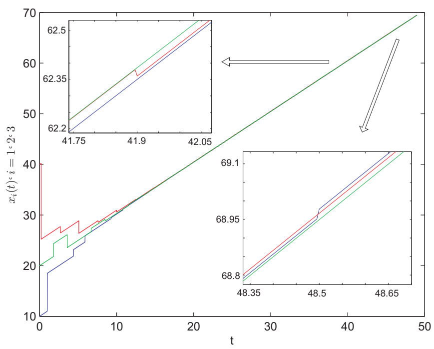


Figure 7.8: *Time displays of three clocks with asymmetric delays in an undirected network.*

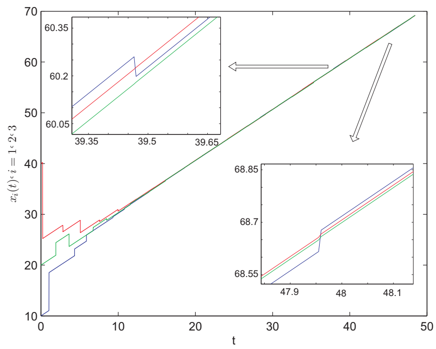


Figure 7.9: *Time displays of three clocks with time-varying delays in an undirected network.*

in Fig. 7.4. The time delays are $d_{12} = 0.2$, $d_{21} = 0.3$, $d_{23} = 0.3$, $d_{32} = 0.4$, $d_{13} = 0.4$, and $d_{31} = 0.3$. Thus the nominal delays for the three pair of clocks $(1, 2)$, $(2, 3)$, and $(1, 3)$ are $\bar{D}_{12} = 0.25$, $\bar{D}_{23} = 0.35$, and $\bar{D}_{13} = 0.35$ respectively. We set the idling times to be $w_i = 0.5$, $i = 1, \ldots, 6$, and the initial time displays of the three clocks to be $[x_1(0), x_2(0), x_3(0)]^T = [10, 40, 20]^T$. The simulation results of the evolution of the displays of the three clocks are shown in Fig. 7.8, from which one can find that the synchronization errors are bounded in a small range without converging to some constants.

When we rerun the simulations for the case when the delays take random values in the intervals $d_{12} \in [0.15, 0.25]$, $d_{21} \in [0.2, 0.4]$, $d_{23} \in [0.2, 0.4]$, $d_{32} \in [0.3, 0.5]$, $d_{13} \in [0.3, 0.5]$, and $d_{31} \in [0.2, 0.4]$. The expected nominal delays are still the same as above. From Fig. 7.9 one can find that the clock synchronization errors are still bounded in a small range.

7.4.3. Example. (*Strongly connected networks*) We consider four clocks with the same skew in a strongly connected graph $\mathbb{G} = \{\mathcal{V}, \mathcal{E}\}$ with $\mathcal{V} = \{1, 2, 3, 4\}$ and $\mathcal{E} = \{(1, 2), (2, 3), (3, 1), (2, 4), (4, 1)\}$ as shown in Fig. 7.10. Let $\mathbb{G}_1 = \{\mathcal{V}_1 = \{1, 2, 3\}, \mathcal{E}_1 = \{(1, 2), (2, 3), (3, 1)\}\}$ and $\mathbb{G}_2 = \{\mathcal{V}_2 = \{1, 2, 4\}, \mathcal{E}_2 = \{(1, 2), (2, 4), (4, 1)\}\}$. Then $\mathbb{G}_1$ and $\mathbb{G}_2$ are both directed ring graphs and $\mathcal{V}_1 \cup \mathcal{V}_2 = \mathcal{V}$. The time delays are $d_{12} = d_{24} = d_{41} = 0.2$, $d_{23} = 0.3$, and $d_{31} = 0.4$. The nominal delay for the directed ring graph $\mathbb{G}_1$ is $\bar{D}_1 = \frac{1}{3}(d_{12} + d_{23} + d_{31}) = 0.3$ and the nominal delay for $\mathbb{G}_2$ is $\bar{D}_2 = \frac{1}{3}(d_{12} + d_{24} + d_{41}) = 0.2$. We set the idling times to be $w_i = 0.5$ and the initial time displays of the four clocks to be $[x_1(0), x_2(0), x_3(0), x_4(0)]^T = [10, 40, 20, 30]^T$. The simulation results of the evolution of the displays of the four clocks are shown in Fig. 7.11, from which one can find that the synchronization errors oscillate among

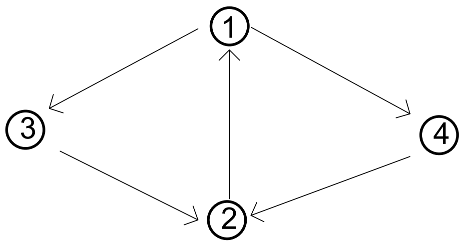


Figure 7.10: *The graph topology of a general strongly connected graph.*

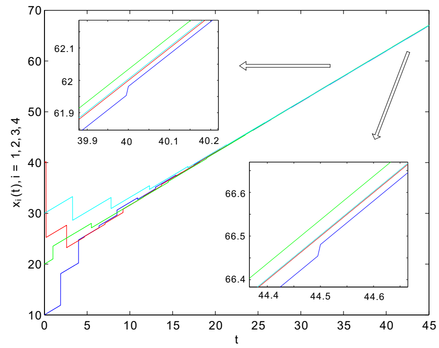


Figure 7.11: *Time displays of four clocks in a strongly connected network.*

several values.

7.5 Conclusion

We have presented explicit expressions for the asymptotic synchronization errors between two interconnected clocks, and expanded the results to larger networks with directed ring topologies, connected undirected topologies, and general strongly connected directed topologies respectively. The obtained synchronization errors complement the impossibility results for clock synchronization in the literature. Deterministic time delays have been considered in this chapter. Questions on the determination of clock synchronization errors in realistic data networks are still open, where the time delays are random.

Chapter 8

Conclusions and future research

In this chapter, we summarize our work in this thesis and give recommendations for future research.

8.1 Concluding remarks

This thesis has investigated distributed algorithms for multi-agent systems and clock synchronization in distributed networks. To aid the analysis of the convergence of these distributed algorithms, we have reviewed some classical and recent results on the convergence of backward products of stochastic matrices. A new necessary and sufficient condition has been given by making use of the Sarymsakov class of stochastic matrices, which we have reexamined and made a connection to those well-understood SIA matrices.

The set of scrambling stochastic matrices has been used to construct a sufficient condition to guarantee the convergence of an asynchronous coordination algorithm based on the convergence results on products of stochastic matrices in Chapter 3.

We have generalized discussions on distributed algorithms to settings where the couplings between pairs of agents in a network could be positive or negative in Chapter 4. By making use of the notion of structural balance, sufficient conditions have been constructed to guarantee that the states of the agents polarize or reach an agreement of zero value. The systems studied here can be used to model the opinion dynamics in social networks, which are often structurally balanced and can be divided into two opposing factions.

Chapter 5 has investigated three different algorithms that lead to n-cluster synchronization in multi-agent systems. Some sufficient conditions and/or necessary conditions have been constructed for systems with different agent self-dynamics, with delay or having negative couplings. The three models presented here are just examples of different approaches towards cluster synchronization and much richer cluster synchronization behaviors in natural and man-made systems require further identification of more such models. More insights have been gained by jointly studying the controllability problem and the cluster synchronization problem of multi-agent systems. Those multi-agent networks that are uncontrollable in finite time tend to realize cluster synchronization as time goes to infinity.

We have explored clock synchronization problem in distributed networks and presented explicit expressions for the asymptotic synchronization errors between two interconnected clocks, and expanded the results to larger networks with directed ring topologies, connected undirected topologies, and general strongly connected directed topologies respectively. The obtained synchronization errors complement the impossibility results for clock synchronization in the literature.

8.2 Recommendations for future research

We have seen that the theory on products of stochastic matrices is fundamental in establishing the effectiveness of distributed coordination algorithms. This is closely related to the theory on ergodicity of non-homogeneous Markov chains. (See [40] for detailed information.) One possible way to study the ergodicity of a chain of stochastic matrices is to check the total information flow between two disjoint sets of agents in a network over time, where the notions of infinity flow and absolute infinite flow are used to attack this problem [96, 95]. It has been shown that the absolute infinite flow property of a chain of stochastic matrices is necessary, but not sufficient for the ergodicity of this chain. People are still making efforts to find out what additional conditions are needed to make a chain of stochastic matrices ergodic.

Chapter 3 has studied the asynchronous implementation of a distributed coordination algorithm. It is also interesting to consider the case when the agents update asynchronously in distributed algorithms in the presence of positive and negative couplings and also in clustering algorithms. Besides the asynchronous events that may arise in practical situations, other constraints such as time delays, quantized information, noisy measurements and so on, are also important issues that one should take into account when dealing with realistic systems. Another active research area in coordination algorithms is the analysis of the convergence speed of these algorithms [74], which is an important characterization of the performance of these algorithms. Some authors have proposed algorithms that achieve finite-time convergence in a network [27] and some have proposed distributed algorithms that make use of the states of the agents in previous steps to accelerate the convergence process [18, 75]. We have obtained some preliminary results on accelerating consensus by removing specific edges in a network from simulations and further work will follow this line and seek to provide theoretical proofs.

We have generalized distributed algorithms for multi-agent systems by incorporating negative couplings between agents in the interaction graph topologies in Chapter 4 and proposed an algorithm that might lead to clustering phenomena by making use of negative couplings in Chapter 5. Negative coupling, that represents competitive or conflicting relationships between a pair of agents in a network and modeled by an edge with negative weight in a graph, definitely deserves further investigation. Numerical computation has found out that in some cases, when some pairs of agents are conflicting, consensus in a network can be reached with a much faster speed [111]. Said differently, conflict sometimes accelerates the reaching of an agreement among agents. However, there is almost no rigorous mathematical analysis on how an agreement is reached in the presence of both positive and negative couplings. Note that in Chapter 4, agreement can be reached when two opposing factions exist, but the agreed value is trivially zero. It is interesting to find out when a nontrivial agreed value can be achieved in the presence of negative couplings. Preliminary work in this direction has been given in [31]. The results there include the matrix-tree theorem as a special case, which may have great potential to be further developed to carry out the analysis for the dynamical behavior of the system concerned. This may lead to some necessary and sufficient conditions for reaching an agreement in the presence of negative couplings and serve as the first step of explaining how conflicts may sometimes accelerate the process of reaching agreement.

Another issue that deserves consideration is the case when the interaction graph topologies are not only purely dependent on time but also on the system state, since in practical situations, whether a link between a pair of agents exists or not probably depends on the relative differences between the two agents [85, 43]. For example, in social networks, if the state of an agent denotes the opinion of an individual towards a subject, then a link between a pair of individuals could vanish or establish as the difference between their opinions varies, which induces state-dependent interaction topologies [43]. Thus the dynamics of the agents and the interaction graph topologies are interacting with each other and it brings great challenges to analyze the dynamical behavior of the system. The convergence results on reaching an agreement or splitting into clusters in a network in the present thesis and also in most of the literature, are derived under proper connectivity assumptions on the graph topologies. However, in general, it is difficult to check whether the connectivity assumptions can be met as the system evolves given an initial condition of the system, when the graph topologies are depending on the state of the system. Recent work on opinion dynamics models has been devoted to this research direction [11, 12, 68]. There are still a number of open problems in these opinion dynamics models with state-dependent interaction graphs especially in the models with asymmetric confidence bound [68].

It has been seen in Chapter 4 that the states of the agents in a network evolve into two opposite values in a structurally balanced network. Structural balance theory is a static theory, and people are also trying to find out how structural balance dynamically arises in a social network. Establishing a satisfactory dynamical model describing the emergence of structural balance is challenging. Present models are focused on networks with all-to-all connections [64] and models for other network topologies are not available. An extension of structural balance of a network is the clustering structure, in which the network can be split into more than two clusters such that each positive connection links two agents of the same cluster and each negative connection links agents from different clusters [28]. A proper dynamical model that can describe the emergence of clustering in social networks is still not available. Another interesting problem is to find out how the opinions evolve in a network that has a clustering structure and to see whether they will finally evolve into several clusters.

Bibliography

[1] D. Aeyels and F. De Smet. A mathematical model for the dynamics of clustering. *Physica D*, 237:2517–2530, 2009.

[2] C. Altafini. Dynamics of opinion forming in structurally balanced social networks. *PLoS ONE*, 7(6):e38135, 2012.

[3] C. Altafini. Consensus problems on networks with antagonistic interactions. *IEEE Transactions on Automatic Control*, 58:935–946, 2013.

[4] B. D. O. Anderson, C. Yu, B. Fidan, and J. Hendrickx. Rigid graph control architectures for autonomous formations. *IEEE Control Systems Magazine*, 28:48–63, 2008.

[5] D. Angeli and P. Bliman. Convergence speed of unsteady distributed consensus: Decay estimate along the setting spanning-trees. *SIAM Journal on Control and Optimization*, 48:1–32, 2009.

[6] J. M. Anthonisse and H. Tijms. Exponential convergence of products of stochastic matrices. *Journal of Mathematical Analysis and Applications*, 59:360–364, 1977.

[7] A. Arenas, A. Diaz-Guilera, J. Kurths, Y. Moreno, and C. Zhou. Synchronization in complex networks. *Physics Reports*, 469:93–153, 2008.

[8] K. Arvind. Probabilistic clock synchronization in distributed systems. *IEEE Transactions on Parallel and Distributed Systems*, 5:474–487, 1994.

[9] A-L. Barabasi and R. Albert. Emergence of scaling in random networks. *Science*, 76:509–512, 1999.

[10] D. P. Bertsekas and J. N. Tsitsiklis. *Parallel and Distributed Computation*. Athena Scientific, Nashua, 1997.

[11] V. D. Blondel, J. M. Hendrickx, and J. N. Tsitsiklis. On Krause's multi-agent consensus model with state-dependent connectivity. *IEEE Transactions on Automatic Control*, 54:2586–2597, 2009.

[12] V. D. Blondel, J. M. Hendrickx, and J. N. Tsitsiklis. Continuous-time average-preserving opinion dynamics with opinion-dependent communications. *SIAM Journal on Control and Optimization*, 48:5214–5240, 2010.

[13] F. Bullo, J. Cortes, and S. Martinez. *Distributed Control of Robotic Networks*. Princeton University Press, Princeton, 2009.

[14] J. Cao. Periodic solutions and exponential stability in delayed cellular neural networks. *Physical Review E*, 60:3244–3248, 1999.

[15] M. Cao, A. S. Morse, and B. D. O. Anderson. Agreeing asynchronously. *IEEE Transactions on Automatic Control*, 53:1826–1838, 2008.

[16] M. Cao, A. S. Morse, and B. D. O. Anderson. Reaching a consensus in a dynamically changing environment: A graphical approach. *SIAM Journal on Control and Optimization*, 47:575–600, 2008.

[17] M. Cao, A. S. Morse, and B. D. O. Anderson. Reaching a consensus in a dynamically changing environment: Convergence rates, measurement delays and asynchronous events. *SIAM Journal on Control and Optimization*, 47:601–623, 2008.

[18] M. Cao, D. A. Spielman, and E. M. Yeh. Accelerated gossip algorithms for distributed computation. In *Proc. of the 44th Annual Allerton Conference*, pages 952–959, 2006.

[19] M. Cao, S. Zhang, and M. K. Camlibel. A class of uncontrollable diffusively coupled multi-agent systems with multi-chain topologies. *IEEE Transactions on Automatic Control*, 58:465–469, 2013.

[20] D. M. Cardoso, C. Delorme, and P. Rama. Laplacian eigenvectors and eigenvalues and almost equitable partitions. *European Journal of Combinatorics*, 28:665–673, 2007.

[21] R. Carli, F. Bullo, and S. Zampieri. Quantized average consensus via dynamic coding/decoding schemes. *International Journal of Robust and Nonlinear Control*, 20:156–175, 2010.

[22] D. Cartwright and F. Harary. Structural balance: a generalization of Heider's theory. *Psychological Review*, 63:277–292, 1956.

[23] F. Ceragioli, C. De Persis, and P. Frasca. Discontinuities and hysteresis in quantized average consensus. *Automatica*, 47:1916–1928, 2011.

[24] G. Chen, X. Wang, and X. Li. *Introduction to Complex Networks: Models, Structures and Dynamics*. Higher Education Press, Beijing, 2012.

[25] F. R. K. Chung. *Spectral Graph Theory*. Number 92 in CBMS Regional Conference Series in Mathematics, American Mathematical Society, 1997.

[26] D. Coppersmith and C. W. Wu. Conditions for weak ergodicity of inhomogeneous Markov chains. *Statistics and Probability Letters*, 78:3082–3085, 2008.

[27] J. Cortés. Finite-time convergent gradient flows with applications to network consensus. *Automatica*, 42:1993–2000, 2006.

[28] J. A. Davis. Clustering and structural balance in graphs. *Human Relations*, 20:181–187, 1967.

[29] M. H. DeGroot. Reaching a consensus. *Journal of the American Statistical Association*, 69:118–121, 1974.

[30] R. Diestel. *Graph Theory*. Springer-Verlag New York, Inc., 1997.

[31] A-L. Do, S. Boccaletti, and T. Gross. Graphical notation reveals topological stability criteria for collective dynamics in complex networks. *Physical Review Letters*, 108:194102, 2012.

[32] A. S. Dolby and T. C. Grubb. Benefits to satellite members in mixed species foraging groups: an experimental analysis. *Animal Behaviour*, 56:501–509, 1998.

[33] R. D. Driver. *Ordinary and Delay Differential Equations*. Springer-Verlag, New York, 1977.

[34] J. C. Eidson. *Measurement, Control, and Communication Using IEEE 1588*. Springer-Verlag, New York, 2006.

[35] G. Facchetti, G. Iacono, and C. Altafini. Computing global structural balance in large-scale signed social networks. *Proceedings of the National Academy of Sciences*, 108:20953–20958, 2011.

[36] N. Freris, H. Kowshik, and P. R. Kumar. Fundamentals of large sensor networks: Connectivity, capacity, clock and computation. *Proceedings of the IEEE*, 98:1828–1846, 2010.

[37] N. M. Freris, S. R. Graham, and P. R. Kumar. Fundamental limits on synchronizing clocks over networks. *IEEE Transactions on Automatic Control*, 55:1352–1364, 2011.

[38] A. Gerbaud. Spectra of generalized compositions of graphs and hierarchical networks. *Discrete Mathematics*, 310:2824–2830, 2010.

[39] C. Godsil and G. Royle. *Algebraic Graph Theory*. Springer-Verlag, New York, 2001.

[40] J. Hajnal. Weak ergodicity in non-homogeneous Markov chains. *Proceedings of Cambridge Philosophy Society*, 54:233–246, 1958.

[41] F. Harary. On the notion of balance of a signed graph. *Michigan Mathematical Journal*, 2:143–146, 1953-1954.

[42] D. J. Hartfiel. *Nonhomogeneous Matrix Products*. World Scientific, Singapore, 2002.

[43] R. Hegselmann and U. Krause. Opinion dynamics and bounded confidence: Models, analysis and simulation. *Journal of Artificial Societies and Social Simulation*, 5:1–24, 2002.

[44] R. A. Horn and C. R. Johnson. *Matrix Analysis*. Cambridge University Press, Cambridge, U.K., 1985.

[45] Y. Hou, J. Li, and Y. Pan. On the Laplacian eigenvalues of signed graphs. *Linear and Multilinear Algebra*, 51:21–30, 2003.

[46] IEEE Instrumentation and Measurement Society. 1588 IEEE standar for a precision clock synchronization protocol for networked measurement and control systems. In *IEEE Standards*. 2002.

[47] A. Jadbabaie, J. Lin, and A. S. Morse. Coordination of groups of mobile autonomous agents using nearest neighbor rules. *IEEE Transactions on Automatic Control*, 48:985–1001, 2003.

[48] E. Serpekin K.-L. Noh, Q. M. Chaudhari and B. W. Suter. Novel clock phase offset and skew estimation using two-way timing message exchanges for wireless sensor networks. *IEEE Transactions on Communications*, 55:766–777, 2007.

[49] R. E. Kalman. Mathematical description of linear dynamical systems. *Journal of Society of Industrial and Applied Mathematics*, 1:152–192, 1963.

[50] A. Kashyap, T. Basar, and R. Srikant. Quantized consensus. *Automatica*, 43:1192–1203, 2007.

[51] L. Kocarev and U. Parlitz. Generalized synchronization, predictability, and equivalence of unidirectionally coupled dynamical systems. *Physical Review Letters*, 76:1816–1819, 1996.

[52] L. Lamport. Time, clocks, and the ordering of events in a distributed system. *Communications of the ACM*, 21:558–565, 1978.

[53] Q. Li and D. Rus. Global clock synchronization in sensor networks. *IEEE Transactions on Computers*, 55:214–226, 2006.

[54] J. Lin, A. S. Morse, and B. D. O. Anderson. The multi-agent rendezvous problem–part 2: The asynchronous case. *SIAM Journal on Control and Optimization*, 46:2120–2147, 2007.

[55] Z. Lin. *Distributed Control and Analysis of Coupled Cell Systems*. VDM-Verlag, Saarbrucken, Germany, 2008.

[56] C. Lorand and P. H. Bauer. Distributed discrete-time systems with synchronization errors: Models and stability. *IEEE Transactions on Circuits and Systems II*, 52:208–212, 2005.

[57] C. Lorand and P. H. Bauer. Clock synchronization errors in circuits: Models, stability and fault detection. *IEEE Transactions on Circuits and Systems I*, 53:2299–2305, 2006.

[58] Y. Lou and Y. Hong. Controllability analysis of multi-agent systems with directed and weighted interconnection. *International Journal of Control*, 85:1486–1496, 2012.

[59] W. Lu, T. Chen, and G. Chen. Synchronization analysis of linearly coupled systems described by differential equations with a coupling delay. *Physica D*, 221:118–134, 2006.

[60] W. Lu, B. Liu, and T. Chen. Cluster synchronization in networks of coupled nonidentical dynamical systems. *Chaos*, 20:013120, 2010.

[61] J. Lundelius and N. Lynch. A new fault-tolerant algorithm for clock synchronization. In *Proceedings of the third ACM Symposium on Principles of Distributed Computing*, pages 75–88, 1984.

[62] J. Lundelius and N. Lynch. An upper and lower bound for clock synchronization. *Information and Control*, 62:190–204, 1984.

[63] S. Martini, M. Egerstedt, and A. Bicchi. Controllability analysis of multi-agent systems using relaxed equitable partitions. *International Journal of Systems, Control and Communications*, 2:100–121, 2010.

[64] S. A. Marvel, J. Kleinberg, R. D. Kleinberg, and S. H. Strogatz. Continuous-time model of structural balance. *Proceedings of the National Academies of Sciences*, 108:1771–1776, 2011.

[65] R. Merris. Laplacian matrices of graphs: A survey. *Linear Algebra and its Appliacations*, 197-198:143–176, 1994.

[66] M. Mesbahi and M. Egerstedt. *Graph Theoretic Methods in Multiagent Networks*. Princeton University Press, 2010.

[67] D. L. Mills. Internet time synchronization: The network time protocol. *IEEE Transactions on Communications*, 39:1482–1493, 1991.

[68] A. Mirtabatabaei and F. Bullo. Opinion dynamics in heterogeneous networks: Convergence conjectures and theorems. *SIAM Journal on Control and Optimization*, 50:2763–2785, 2012.

[69] L. Moreau. Stability of multiagent systems with time-dependent communication links. *IEEE Transactions on Automatic Control*, 50:169–182, 2005.

[70] T. Oguchi, H. Nijmeijer, and T. Yamamoto. Synchronization in networks of chaotic systems with time-delay coupling. *Chaos*, 18:037108, 2008.

[71] R. Olfati-Saber. Flocking for multi-agent dynamic systems: Algorithms and theory. *IEEE Transactions on Automatic Control*, 51:401–420, 2006.

[72] R. Olfati-Saber, J. A. Fax, and R. M. Murray. Consensus and cooperation in networked multi-agent systems. *Proceedings of the IEEE*, 95:215–233, 2007.

[73] R. Olfati-Saber and R. M. Murray. Consensus problems in networks of agents with switching topology and time-delays. volume 49, pages 1520–1533, 2004.

[74] A. Olshevsky and J. T. Tsitsiklis. Convergence speed in distributed consensus and averaging. *SIAM Journal on Control and Optimization*, 48:33–55, 2009.

[75] B. N. Oreshkin, M. J. Coates, and M. G. Rabbat. Optimization and analysis of distributed averaging with short node memory. *IEEE Transactions on Signal Processing*, 58:2850–2865, 2010.

[76] S. Palchaudhuri, A. K. Saha, and D. B. Johnson. Adaptive clock synchronization in sensor networks. In *Proceedings of the third ACM/IEEE International Conference on Information Processing in Sensor Networks*, 2004.

[77] L. M. Pecora and T. L. Carroll. Synchronization in chaotic systems. *Physical Review Letters*, 64:821–824, 1990.

[78] A. Y. Pogromsky. A partial synchronization theorem. *Chaos*, 18:037107, 2008.

[79] A. Rahmani, M. Ji, M. Mesbahi, and M. Egerstedt. Controllability of multi-agent systems from a graph-theoretic perspective. *SIAM Journal on Control and Optimization*, 48:162–186, 2009.

[80] W. Ren and R. Beard. *Distributed consensus in Multi-Vehicle Cooperative Control*. Springer-Verlag, London, 2008.

[81] W. Ren and R. W. Beard. Consensus seeking in multiagent systems under dynamically changing interaction topologies. *IEEE Transactions on Automatic Control*, 50:655–661, 2005.

[82] W. Ren and Y. Cao. *Distributed coordination of multi-agent networks: Emergent problems, models, and issues*. Springer-Verlag, London, 2011.

[83] C. W. Reynolds. Flocks, herds and schools: A distributed behavioral model. *Computer Graphics*, 21:71–87, 1987.

[84] K. Romer. Time synchronization in Ad Hoc networks. In *Proceedings of the second ACM International Symposium on Mobile Ad Hoc Networking and Computing*, 2001.

[85] A. Sarlette. *Geometry and symmetries in coordination control*. PhD Dissertation, 2009.

[86] T. A. Sarymsakov. Inhomogeneous Markov chains (in russian). *Teor. Verojatnost. i Primen.*, 6:194–201, 1961.

[87] L. Scardovi and R. Sepulchre. Synchronization in networks of identical linear systems. *Automatica*, 45:2557–2562, 2009.

[88] L. Schenato and F. Fiorentin. Average Timesynch: A consensus-based protocol for clock synchronization in wireless sensor networks. *Automatica*, 47:1878–1886, 2011.

[89] E. Seneta. Coefficients of ergodicity: Structure and applications. *Advances in Applied Probability*, 11:576–590, 1979.

[90] E. Seneta. *Non-negative Matrices and Markov Chains.* Springer-Verlag, 2006. 2nd edition.

[91] F. De Smet and D. Aeyels. Clustering in a network of non-identical and mutually interacting agents. *Proceedings of the Royal Society A*, 495:745–768, 2009.

[92] T. K. Srikanth and S. Toueg. Optimal clock synchronization. *Journal of the ACM*, 34:626–645, 1987.

[93] S. H. Strogatz. *SYNC: The emerging science of spontaneous order.* Hyperion, New York, 2003.

[94] W. Su and F. Akyildiz. Time-diffusion synchronization protocol for wireless sensor networks. *IEEE Transactions on Networking*, 13:384–397, 2005.

[95] B. Touri and A. Nedic. On ergodicity, infinite flow and consensus in random models. *IEEE Transactions on Automatic Control*, 56:1593–1605, 2011.

[96] B. Touri and A. Nedic. On backward product of stochastic matrices. *Automatica*, 48:1477–1488, 2012.

[97] T. Vicsek, A. Czirok, E. Ben-Jacob, I. Cohen, and O. Shochet. Novel type of phase transition in a system of self-driven particles. *Physical Review Letters*, 75:1226–1229, 1995.

[98] S. Wasserman and K. Faust. *Social Network Analysis: methods and applications.* Cambridge University Press, Cambridge, U.K., 1994.

[99] D. J. Watts and S. H. Strogatz. Collective dynamics of 'small-world' networks. *Nature*, 393:440–442, 1998.

[100] H. R. Wilson and J. D. Cowan. Excitator and inhibitory in localized populations of model neurons. *Biophysical Journal*, 12:1–24, 1972.

[101] J. Wolfowitz. Products of indecomposable, aperiodic, stochastic matrices. *Proceedings of the American Mathematical Society*, 14:733–737, 1963.

[102] C. W. Wu. Synchronization in arrays of coupled nonlinear systems with delay and nonreciprocal time-varying coupling. *IEEE Transactions on Circuits and Systems II*, 52:282–286, 2005.

[103] C. W. Wu and L. O. Chua. Synchronization in an array of linearly coupled dynamical systems. *IEEE Transactions on Circuits and Systems I*, 42:430–447, 1995.

[104] W. Wu, W. Zhou, and T. Chen. Cluster synchronization of linearly coupled complex networks under pinning control. *IEEE Transactions on Circuits and Systems I*, 56:829–839, 2009.

[105] W. Xia and M. Cao. Cluster synchronization algorithms. In *Proc. of the 2010 American Control Conference*, pages 6513–6518, 2010.

[106] W. Xia and M. Cao. Clustering in diffusively coupled networks. *Automatica*, 47:2395–2405, 2011.

[107] W. Xia and M. Cao. Synchronizing clocks in distributed networks. In *Proc. of the 50th IEEE Conference on Decision and Control and 2011 European Control Conference*, pages 777–782, 2011.

[108] W. Xia and M. Cao. Cluster synchronization and controllability of complex multi-agent networks. In *Proc. of the 2012 IEEE International Symposium on Circuits and Systems*, pages 165–168, 2012.

[109] W. Xia and M. Cao. Sarymsakov matrices and their application in coordinating multi-agent systems. In *Proc. of the 31st Chinese Control Conference*, pages 6321–6326, 2012.

[110] F. Xiao and L. Wang. Asynchronous consensus in continuous-time multi-agent systems with switching topology and time-varying delays. *IEEE Transactions on Automatic Control*, 53:1804–1816, 2008.

[111] L. Xiao and S. Boyd. Faster linear iterations for distributed averaging. *Systems and Control Letters*, 53:65–78, 2004.

[112] J. Yu and L. Wang. Group consensus of multi-agent systems with undirected communication graphs. In *Proc. of the 7th Asian Control Conference*, pages 105–110, 2009.

[113] S. Zhang, M. K. Camlibel, and M. Cao. Controllability of diffusively-coupled multi-agent systems with general and distance regular coupling topologies. In *Proc. of the 50th IEEE Conference on Decision and Control and 2011 European Control Conference*, pages 759–764, 2011.

[114] J. Zhou, L. Xiang, and Z. Liu. Global synchronization in general complex delayed dynamical networks and its applications. *Physica A*, 385:729–742, 2007.

Summary

In the control community, the study on distributed control of multi-agent systems has received considerable attention in recent years due to their broad applications in sensor networks, robotic teams, and so on. This is also motivated by the growing interest in understanding the collective group behaviors in natural, social and engineered networks. The architecture of these complex networks is distributed or decentralized in character. Each individual agent in the complex multi-agent system does not have global information of all the agents; instead, each agent only interacts with its neighbors, receives limited information from them, and takes actions based on the local information. Huge efforts have been devoted to investigate the interplay between the individual agent dynamics and the network structure in order to understand the emergence of collective behaviors. This thesis is concerned with distributed algorithms for interacting autonomous agents. We study several distributed algorithms that drive a group of agents to reach an agreement on the value of a variable of common interest or to split into two or more clusters. The clock synchronization problem in distributed networks with communication time delays is also discussed.

We reexamine a subclass of *stochastic matrices*, the *Sarymsakov class* of stochastic matrices and explore its relationship with other well-studied classes of stochastic matrices. The classical conditions scattered in the literature for the convergence of products of stochastic matrices are reviewed and a new necessary and sufficient condition is then proposed by making use of the Sarymsakov matrices. These convergence results serve as fundamental tools for the analysis of distributed coordination algorithms for multi-agent systems. They are applied to solve an asynchronous implementation problem of a distributed coordination algorithm that causes a group of agents to reach an agreement.

By employing the *structural balance* theory from social networks study, we study

distributed algorithms in the presence of positive and negative couplings. These models differ from most of those investigated in the literature, which only consider positive couplings in networks. Sufficient conditions are constructed to show when the state of the system polarizes or converges to an agreed value of zero.

To better understand the *clustering behavior* emerging in natural and man-made systems, three different mechanisms that may lead to the clustering behavior are proposed and analyzed. Some sufficient conditions and/ or necessary conditions are constructed for systems with different agent self-dynamics, with delay or having negative couplings. These clustering mechanisms are examples of different mechanisms leading to clustering phenomena. More mechanisms may be identified after gaining insight into the clustering behavior in natural and man-made systems. Furthermore, by jointly studying the controllability problem and the cluster synchronization problem of multi-agent systems, it is shown that those multi-agent networks that are uncontrollable in finite time tend to realize cluster synchronization as time goes to infinity.

Another issue that is addressed in the thesis is the clock synchronization problem in distributed networks with communication time delays. Recently, there have been studies showing the impossibility of clock synchronization in distributed networks with asymmetric time delays. Based on similar models for clocks, we derive explicit expressions for the asymptotic synchronization errors between two interconnected clocks and expand the results to larger networks with directed ring topologies, connected undirected topologies, and general strongly connected directed topologies, respectively.

Samenvatting

In de systeemtheorie en regeltechniek heeft het onderzoek naar gedistribueerde aansturing van multi-agent systemen de afgelopen jaren veel aandacht gekregen door hun brede toepassing in sensor netwerken, robot teams, enzovoort. De motivatie voor dit onderzoek komt ook voort uit de groeiende interesse om het collectieve gedrag van groepen in natuurlijke, sociale en geconstrueerde netwerken beter te begrijpen. De architectuur van dergelijke complexe netwerken heeft een gedistribueerd of gedecentraliseerd karakter. Geen van de individuele agenten heeft kennis van de globale informatie aangaande alle agenten; in plaats daarvan kan elke agent alleen met zijn buren samenwerken, gelimiteerde informatie van hen ontvangen en actie ondernemen op basis van deze lokale informatie. Veel arbeid is geleverd om de wisselwerking tussen de dynamica van de individuele agenten en de structuur van het netwerk te onderzoeken met het doel het ontstaan van collectief gedrag beter te begrijpen. Dit proefschrift behandelt gedistribueerde algoritmes voor samenwerkende autonome agenten. We bestuderen verschillende gedistribueerde algoritmes voor een groep agenten die een overeenkomst willen bereiken over een variabele van gezamenlijke interesse of de groep splitst in twee of meer clusters. Het kloksynchronisatie probleem in gedistribueerde netwerken met tijdsvertragingen als gevolg van communicatie wordt ook behandeld.

We heroverwegen een subklasse van *stochastische matrices*, de zogenoemde *Sarymsakov klasse* van stochastische matrices, en verkennen de relatie met andere bekende klassen van stochastische matrices. Klassieke voorwaarden uit de literatuur aangaande de convergentie van producten van stochastische matrices zijn heroverwogen en een nieuwe noodzakelijke en voldoende voorwaarde zijn voorgesteld door gebruik te maken van de Sarymsakov matrices. Deze convergentie resultaten worden gebruikt voor de analyse van gedistribueerde algoritmes voor multi-agent systemen. De voorwaarden zijn toegepast voor de oplossing van een asynchroon implementatieprobleem

voor een gedistribueerd coördinatie algoritme, wat ervoor zorgt dat een groep agenten overeenstemming bereikt.

Gebruik makend van de theorie van *structurele balans*, welke zijn oorsprong vindt in de studie van sociale netwerken, onderzoeken we gedistribueerde algoritmes in de aanwezigheid van positieve en negatieve koppelingen. Deze modellen verschillen van reeds onderzochte modellen, welke alleen positieve koppelingen beschouwen. Voldoende voorwaarden zijn opgesteld, welke laten zien wanneer de toestand van het systeem polariseert, dan wel convergeert naar een afgesproken waarde van nul.

Om het *clustering gedrag*, wat verschijnt in natuurlijke en kunstmatige systemen, beter te begrijpen zijn drie verschillende mechanismes die tot clustering gedrag kunnen leiden voorgesteld en geanalyseerd. Een aantal voldoende voorwaarden en/of noodzakelijke voorwaarden zijn geconstrueerd voor systemen met; verschillende zelf-dynamica van de agenten, tijdsvertragingen, of negatieve koppelingen. Deze clustering mechanismes zijn voorbeelden van verschillende mechanismes die leiden tot clustering verschijnselen. Meer mechanismes kunnen worden onderscheiden nadat meer inzicht is verkregen in het clustering gedrag in natuurlijke en kunstmatige systemen. Bovendien, door het beheersbaarheid probleem en het cluster synchronisatie probleem gezamenlijk te bestuderen, is aangetoond dat die meerdere-agent systemen die niet beheersbaar zijn in eindige tijd, cluster synchronisatie realiseren wanneer de tijd naar oneindig gaat.

Een ander onderwerp in dit proefschrift is het klok synchronisatie probleem in gedistribueerde netwerken met communicatie tijdsvertragingen. Recentelijk zijn er onderzoeken geweest die aantonen dat klok synchronisatie in gedistribueerde netwerken met asymmetrische tijdsvertragingen onmogelijk is. Op basis van vergelijkbare modellen voor klokken, leiden wij expliciete uitdrukkingen af voor de asynchrone synchronisatiefouten tussen twee verbonden klokken en breiden deze resultaten uit naar grotere netwerken met respectievelijk gerichte ring topologiën, verbonden niet-gerichte topologiën, en algemene sterk verbonden gerichte topologiën.