COOPERATIVE CONTROL OF MULTI-AGENT SYSTEMS WITH INFORMATION FLOW CONSTRAINTS

A Dissertation

Submitted to the Graduate School
of the University of Notre Dame
in Partial Fulfillment of the Requirements
for the Degree of

Doctor of Philosophy

by

Lei Fang, B.S., M.S.

Panos J. Antsaklis, Director

Graduate Program in Electrical Engineering
Notre Dame, Indiana
July 2006
Abstract

by

Lei Fang

Motivated by the emergence of new applications of networked, large-scale multi-agent systems, we study several important and related problems in this area: asynchronous consensus (agreement) problems, formation tracking control, and switched linear control. The overall goal is to understand special challenges raised from information constraints and to develop robust control strategies which improve the stability and performance of such systems.

We start our investigation with asynchronous consensus problems for discrete time multi-agent systems. In this setup, a number of agents update their states asynchronously by using (possibly outdated) information from their neighbors in order to reach an agreement regarding a certain quantity of interest. Under fixed interaction topologies, we show that consensus can be reached with linear protocols. We further show that consensus is reachable under directional and time-varying topologies with nonlinear protocols. The confluent iteration graph is introduced to incorporate various communication assumptions and it proves to be fundamental in understanding the convergence of consensus processes.

Secondly, we study formation tracking problems which can be stated as multiple vehicles with nonlinear dynamics being required to follow reference trajectories while
keeping a desired inter-vehicle formation pattern in time. We specify formations using the vectors of relative positions of neighboring vehicles and use consensus-based controllers for decentralized formation tracking control. The key idea is to combine consensus-based controllers with the cascaded approach to tracking control, resulting in a group of linearly coupled dynamical systems. We examine the stability properties of the closed loop system using cascaded systems theory and nonlinear synchronization theory. In particular, we identify a link between a property on the graph Laplacian of the information structure and the formation stability.

The last part of the dissertation is dedicated to switched systems, which are natural mathematical models for time-varying topologies among agents and are of theoretical interest on their own. By using tools from convex analysis, we provide a new proof to a necessary and sufficient stability condition for switched systems under arbitrarily switching. The switched Lyapunov function is then combined with Finslers’ Lemma to generate various LMI conditions for control synthesis and performance analysis of switched systems.
To my dear wife Xiaofang and my parents.
3.2.4 A Classification of Consensus Results .......................... 48
3.3 Bidirectional vs. Unidirectional Communication .................... 50
3.4 Iteration Graph and Nonlinear Asynchronous Consensus Protocols 54
  3.4.1 Iteration Graph ........................................ 55
  3.4.2 Nonlinear Asynchronous Consensus Protocols .................. 57

CHAPTER 4: DECENTRALIZED FORMATION TRACKING OF MULTIPLE-
VEHICLE SYSTEMS .................................................... 61
  4.1 Introduction .................................................. 61
  4.2 Preliminaries .................................................. 64
    4.2.1 Tracking Control of Mobile Vehicles ...................... 64
    4.2.2 Formation Graphs ........................................ 67
    4.2.3 Input-to-state Stability on Formation Graphs ............ 68
    4.2.4 Formation Control as Consensus Problems ................ 68
    4.2.5 Synchronization in networks of nonlinear dynamical systems 69
  4.3 Basic Formation Tracking Controller .......................... 70
  4.4 Consensus-based Formation Tracking Controller ................ 72
  4.5 Formation ISS ............................................... 78
  4.6 Conclusions .................................................. 82

CHAPTER 5: STABILIZATION AND PERFORMANCE ANALYSIS FOR
A CLASS OF SWITCHED SYSTEMS ...................................... 85
  5.1 Introduction .................................................. 85
  5.2 A Necessary and Sufficient Condition .......................... 88
    5.2.1 Stability Notions of Switched Linear Systems and Linear Time-
      Varying Systems ........................................... 88
    5.2.2 Equivalence between Stability Concepts .................... 89
    5.2.3 Application of the Equivalence Result for Special Cases .... 93
  5.3 Stability and Performance Analysis: A Switched Lyapunov Function
    Approach ....................................................... 96
    5.3.1 Problem Formulation ...................................... 97
    5.3.2 Stability Analysis ........................................ 98
    5.3.3 Attainable Bounds on Decay Rate .......................... 103
    5.3.4 Input-Output Performance ................................ 107

CHAPTER 6: CONCLUSIONS AND FUTURE WORK ........................ 115
  6.1 A Summary of Contributions .................................... 115
  6.2 Future Research Directions .................................... 116

BIBLIOGRAPHY ....................................................... 118
FIGURES

2.1 The interactions among mobile agents are often asynchronous. ........................................ 27
2.2 The (synchronous) consensus value depends on the initial states and the interaction topologies. ................................................................. 31
2.3 Asynchronous multi-agent systems with different interaction topologies. .............................. 33
2.4 The interaction topology of an asynchronous multi-agent system with five agents. ......................... 36
2.5 Asynchronous consensus value \( p = 1/2 \) and \( D = 0 \) depends on various topology selections made in the course of the computations. .................................. 37
2.6 The histogram of the consensus value \( p = 1/2 \) and \( D = 0 \) is unimodal. ............................... 37
2.7 The histogram of the consensus value \( p = 1/2 \) and \( D = 8 \). ........................................ 38
2.8 Synchronous systems with time varying topologies can be seen as a special case of asynchronous systems. ......................................................... 40

3.1 Possible communication topologies for a three-agent system .............................................. 52
3.2 Interaction topologies of an asynchronous system and its associated iteration graph. ................. 56

4.1 Six vehicles perform a formation tracking task. .......................................................... 62
4.2 Mobile robots and the error dynamics. ............................................................................. 64
4.3 Formation tracking using baseline FTC. The reference vehicle sends to vehicle \( i \) the formation specification \( d_{ri} \) as well as the reference velocities \( v_r \) and \( \omega_r \). ......................................................... 71
4.4 Circular motion of three vehicles with a triangle formation. Initial vehicle postures are: \([-8 - 9 \ 3\pi/5]^T \) for vehicle 1 (denoted as *); \([-15 -20 \ \pi/2]^T \) for vehicle 2 (□); \([-10 -15 \ \pi/3]^T \) for vehicle 3 (⋄). ......................................................... 73
4.5 Control signals \( v \) and \( \omega \) for Virtual vehicle: solid line; Vehicle 1: dotted line; Vehicle 2: dashed line; and Vehicle 3: dot-dash line. .................. 74
4.6 A formation graph with formation specifications on edges .............................................. 78
4.7 Tracking a sinusoidal trajectory in a triangle formation. Initial vehicle postures are: $[12 12 0]^T$ for vehicle 1 (denoted as $*$); $[-15 -20 \pi/4]^T$ for vehicle 2 ($\Box$); $[-10 15 -\pi/4]^T$ for vehicle 3 ($\diamond$).

4.8 Vehicle control signals $v_i$'s and $\omega_i$'s.

4.9 Primitive subgraphs [179]. (a) Cascade interconnection of vehicles; (b) Parallel interconnection; (c) Multiple leader interconnection.

4.10 Cascade connection of ISS systems.
ACKNOWLEDGMENTS

The research that has gone into this dissertation has been thoroughly enjoyable. That enjoyment is largely a result of the interaction that I have had with my supervisor, colleagues and friends.

I feel very privileged to have worked with my supervisor, Panayotis J. Antsaklis. To him I owe a great debt of gratitude for his vision, inspiration and encouragement. His pleasant personality and enthusiasm for everything will continue affecting me for many years to come.

I would like to express my gratitude to Prof. Mark Alber, Prof. Martin Haenggi, and Prof. Paulo Tabuada for serving on my committee. They have taught me a great deal about nonlinear dynamical systems, networking theory, and nonlinear control systems theory that build the cornerstone of this research.

The Department of Electrical Engineering and Networked Embedded Sensing and Control Laboratory have provided an excellent environment for my research. I spent many enjoyable hours with department members and fellow students chatting about my little ideas over a cup of tea. Without this rich environment I doubt that many of my ideas would have come to fruition.

Thanks also to my family who have been extremely understanding and supportive of my studies. I feel very lucky to have a family that shares my enthusiasm for academic pursuits.

Finally I would like to thank my wonderful wife, Xiaofang, who has encouraged me so much over the years. Without her, this is mission impossible.
1.1 Multi-agent Systems

Agents are computational entities that sense and act, and decide on their actions in accordance with some tasks or goals. Agents have become popular in diverse areas from information retrieval, manufacturing management, service robotics, underwater and space exploration, to name a few. Agents can have capabilities like autonomous behavior, interaction with other agents, goal directed actions and adaptation to changes in the environment, making it possible to bring new features to applications.

Although, in many cases, agents can act separately to solve a particular problem, it often happens that a complete system made of several different agents has to be designed to cope with a complex problem. A multi-agent system (MAS) can therefore be defined as a collection of, possibly heterogeneous, autonomous agents which are able to interact with each other or with their environments to solve problems that are beyond the individual capabilities or knowledge of each agent. The characteristics of MASs are that [177]:

- each agent has incomplete information or capabilities for solving the problem and, thus, has a limited viewpoint;
- there is no global system control;
- data are decentralized; and
• computation is asynchronous.

The motivation for the increasing interest in MAS research includes the ability of MASs to do the following:

• to solve problems that are too large for a centralized agent to solve because of resource limitations or the sheer risk of having one centralized system that could be a performance bottleneck or could fail at critical times;

• to provide solutions to problems that can naturally be regarded as a society of autonomous interacting component-agents;

• to provide solutions that efficiently use information sources that are spatially distributed;

• to provide solutions in situations where expertise is distributed;

• to enhance performance along the dimensions of: computational efficiency, reliability, extensibility, robustness, maintainability, responsiveness, flexibility, and reuse.

MASs are rapidly having a critical presence in many application domains. For a general overview of the field, we refer readers to [171, 177, 79, 162, 22]. A taxonomy is provided in [48] that classifies multi-agent systems according to communication, computational and other capabilities. The Special Issue on Advances in Multirobot Systems [4] provides a broad sampling of the research that is currently ongoing in the field of distributed mobile robot systems.

1.1.1 Cooperative Multi-agent Systems

A group of agents may have a global goal to solve together, or they may individually have their own goals to pursue. The particular characteristics here is that these agents must coordinate their actions. The need for coordination arises because agents have to:
• share resources and expertise;
• eliminate conflicts due to their actions, or due to conflicting interests to achieve their goal(s) which otherwise cannot be done;
• improve efficiency by enhancing predictability or by reducing redundancy;
• improve global efficiency.

When a group of dynamic agents share information or tasks to accomplish a common, though perhaps not singular, objective, we call the MAS a cooperative MAS. Examples of cooperative MASs might include: robots operating within a manufacturing cell, unmanned aircraft in search and rescue operations or military surveillance and attack missions, arrays of micro satellites that form a distributed large aperture radar, employees operating within an organization, and software agents. The term agent is most often associated with vehicles capable of physical motion such as robots, automobiles, ships, and aircraft, but the definition extends to any agent concept that exhibits a time dependent behavior.

One mechanism to achieve cooperation is through communication, either explicitly by message passing, or implicitly via observation of other agents’ state. It may be impossible or undesirable for these agents to share all their knowledge all the time. Furthermore, exchanging information may incur a cost associated with the required bandwidth or with the risk of revealing it to competing agents. Assuming that communication may not be reliable adds another dimension of complexity to the problem. Other means to provide additional possibilities for coordination include investigating heuristics for negotiation, applying formal models of possibilities of conflicts and cooperation among agent interests, and developing policies to optimize environmental feedback through reinforcement learning.

We assume that cooperation is being used to accomplish some common purpose that is greater than the purpose of each individual, but we recognize that the indi-
vidual may have other objectives as well, perhaps due to being a member of other groups. This implies that cooperation may assume hierarchical forms as well. The decision-making processes (control) are typically thought to be distributed or decentralized to some degree. For if not, a cooperative system could always be modeled as a single entity. Cooperative MASs may involve task sharing and can consist of heterogeneous agents. A survey of cooperative mobile robotics is presented in [30].

Although cooperative MASs provide many potential advantages, they also present many difficult challenges [162]

- how do agents decompose the goal into subgoals that can then be assigned to individual agents based on their capabilities and access to resources,
- how to develop agent organization (authority relationship) and problem solving protocols (information flow) that enable agents to share results and knowledge in a timely, effective manner,
- how do agents maintain coherence and problem solving focus when locally available information can be incorrect, inconsistent, outdated, etc? How to avoid unstable system behavior while guaranteeing system performance under adversarial conditions?

Solutions to these problems are intertwined. In order to succeed, any significant MAS research endeavor has not only to borrow from the past work in traditional single agent AI (artificial intelligence), but should utilize relevant techniques and results from more traditional fields like economics, sociology, management sciences, etc.

Like most of AI, MAS research have shifted focus from building grand unified theories to developing specialized techniques to address requirements for well-defined problem classes. In this dissertation, we too follow this tread and focus on specific problems faced by cooperative MASs.
1.1.2 Motivation and Scope

During the last decade, technological advances in computation and communication have provided efficient and inexpensive means to compute and share information. This breakthrough facilitates the development of new MASs, that promise increased performance, efficiency and robustness. Networked, large scale MASs are currently been deployed in several fields, from automotive, aerospace, and robotics to wireless networks, and operate successfully at a fraction of the cost of alternative centralized designs.

Motivated by the emergence of these new applications, we study MASs from the engineering perspective. The goal is to improve the effectiveness of a MAS both from the viewpoint of the performance in accomplishing certain tasks and in the robustness and reliability of the system. We consider two important and related problems: asynchronous consensus and formation control.

Asynchronous Consensus A team of asynchronous agents reach consensus on the values of interest. This need stems from the fact that in order for agents to coordinate their behaviors, they need to use some shared knowledge about variables such as direction, speed, time-to-target etc. By asynchronism, we mean each agent updates on its own pace, and uses the most recently received (but possibly outdated) information from other agents.

Formation Control A group of mobile agents either achieve a formation, or move while maintaining a formation, or reconfigure from one formation to another. The control of MASs is greatly simplified when agent’s mission can be executed by means of a formation. Other advantages of formation control include increased feasibility, flexibility, robustness, cost and probability of success. These problems are chosen because they are abstract but still capture the typical characteristics of MASs. Compared with the more traditional applications of control theory, there are funda-
mentally new features introduced in these problems. Ubiquitous are the information flow constraints among agents. For instance, an important aspect of information flow is the communication topology, which determined what information is available for which component at a given time instant. Communication topologies thus affect the system-wide behavior and the behavior in turn affects the communication topologies due to, e.g., the limited communication range between agents.

In this dissertation, we put emphasis on understanding how information constraints creates special challenges for designing the control process and developing robust information exchange strategies which improve stability and performance of MASs. Questions that naturally arise include: Is the system goal reachable only via communication? If yes, can an agent use its knowledge about other agents to reduce its communication frequency to save bandwidth? How much can a properly defined cost be lowered by doing so? The problem becomes different if the communications are not continuous but are spaced periodically in time. How should the period be chosen for system stability or for a cost minimization?

In order to properly characterize the present work, it is worth highlighting the hypotheses underlying the MAS that can be summarized as follows:

- Explicit communication-based coordination: Explicit communication is the intentional transmission and reception of information. It is usually achieved with the help of an underlying communication mechanism such as 802.11 wireless ethernet, infrared serial, or more recently, Bluetooth. The communication in itself has a cost and may be imperfect subject to limited range, noise, delay, or dropout effects. The communication topology therefore may be time-varying, with possible topologies ranging from a complete graph, to a hierarchical (tree) based structure.

- Decentralized coordination: the communication capabilities, combined with the local perception constraint, require that each agent, while interacting with oth-
ers, must rely on local control (hence, the system is decentralized).

- Heterogeneity: the agents may be heterogeneous both from hardware and software viewpoints, and they can usually perform the same tasks but with different performance.

- Asynchronism: the computation may be asynchronous due to decentralization and heterogeneity of the system.

The above assumptions directly lead to the question about how the local interactions should be represented. One natural way is to model local interactions via a graph, in which nodes correspond to individual agents and the presence of an edge between two nodes (agents) signifies that an interaction exists between them. The graph is often called interaction, dependency, communication, or connectivity graph (or topology) [55, 131, 127, 90], and it may be undirected or directed. Communication bandwidths, synchronization constraints, and sensor capabilities affect the performance of any chosen graph. One, therefore, often sees graph theoretical models/tools being used in studying consensus or formation control problems.

In the next section, we introduce asynchronous consensus and formation control problems and identify the work in the cooperative control literature that is closely related to our problems.

1.2 Related Work

Research activity in this area can be classified into two main broad categories: the first one deals with large scale dynamic systems [166] or structured systems [153, 6, 38] all subject to (static) information structure constraints, where the main issue is to reduce the computational load stemming from a centralized approach or to exploit the structure of these systems in order to obtain tractable analysis and control synthesis algorithms; the second category considers the problem of controlling and
coordinating teams of cooperating MASs (e.g., unmanned aerial vehicles, multi-robot systems and sensor networks), where information flow and system dynamics (dynamically) interact. In most cases, the agents are coupled through the task they are trying to accomplish, but are otherwise dynamically decoupled.

1.2.1 Decentralized Control Subject to Information Constraints

For many reasons, decentralization is a natural extension for control strategies with respect to large-scale (interconnected) systems. Desire for autonomy of a fleet of UAVs, for example, may preclude the use of a fixed, ground based centralized control system. This, in turn, may stem from limitations in bandwidth, communication delay, or range that are imposed by the situation.

Of course, there are many other examples for which decentralized control is an apt framework for analysis of large scale systems. Satellite formation flying, planetary exploration, rescue operations, automated highway systems, and distributed sensing problems can all benefit from fundamental understanding of decentralized control and of interconnected systems in general.

Classical Viewpoint

Decentralized control, and more generally, control under information constraints, is known to raise some difficulties not encountered in approaches such as the classical LQ (Linear Quadratic problem), or the frequency or algebraic type control synthesis. An important example was presented by Witsenhausen [195] where it was shown that for quadratic stochastic optimal control of a linear system, subject to a decentralized information constraint called non-classical information, a nonlinear controller can achieve greater performance than any linear controller. Under such a non-classical information pattern the cost function is no longer convex in the controller variables [116, 196], a fact which today has increasing importance. In [196], Witsenhausen
summarized several important results on decentralized control at that time, and gave sufficient conditions under which the problem could be formulated so that the standard Linear-Quadratic-Gaussian (LQG) theory could be applied. Under these conditions, an optimal decentralized controller for a linear system could be chosen to be linear. Ho and Chu [77], in the framework of team theory, defined a more general class of information structures, called partially nested, for which the optimal LQG controller is showed to be linear.

Optimization Methods

Certain decentralized control problems, such as the static team problem of [145], have been proved to be intractable. For particular information structures, the controller optimization problem may have a tractable solution. In particular, it was shown by Voulgaris [191] that nested, chained, hierarchical, delayed interaction and communication, and symmetric systems have this property. The common thread in all of these classes is that by taking an input-output point of view we can characterize all stabilizing controllers in terms of convex constraints in the Youla-Kucera parameter. The $H_2$, $l_1$ and $H_\infty$ disturbance rejection problems are solved in [191]. Recently, Rotkowitz and Lall [153] showed that a property called quadratic invariance is necessary and sufficient for the constraint set to be preserved under feedback. In the case where the plant is stable, this allows the constrained minimum-norm control problem to be reduced to a convex optimization problem. The tractable structures of [196, 77, 191, 7, 52, 66, 143] can all be shown to satisfy this property. This approach was extended to the unstable case in [154]. Also see [161] for the close interplay between structural controller design and so-called multiple-objective control problems in which one imposes structural performance specifications on the closed-loop system. Recently, a series of papers [6, 38, 146, 49] consider control
design for distributed systems, where the controller is to adopt and preserve the distributed spatial structure of the nominal system. Controller design often reduces to the semidefinite programming problems which can be solved efficiently using interior point methods.

In a parametric design setting, it is also worth mentioning the attempt towards the optimal decentralized control problem, which is closely related to the optimal output and reduced-order feedback [2, 94, 158, 193, 61]. The most classical one which has received some attention is the following: in the class of linear decentralized stabilizing feedbacks for linear time invariant systems, find the one which minimizes a given cost function. Some papers gave results working with matrix optimality conditions and classical or ‘specialized’ numerical algorithms borrowed form the field of nonlinear programming [111, 110, 182]. In [182], a convergent version of the Anderson-Moore algorithm for the optimal output feedback problem is applied to a class of optimal decentralized control problems. The algorithm is based on a positive definite approximation of the second-order Taylor-series expansion of the loss function. As a special non-linear and non-convex matrix minimization problem, the optimal static output feedback design problem is solved using trust region method in [92]. Furthermore, the specification of a certain margin of stability for the closed-loop system can be incorporated [156].

Lyapunov Type Methods

In the category of Lyapunov type methods, the concept of vector Lyapunov functions has been widely used for stabilizing control design. The subsystems linked through an interconnected network have associated Lyapunov functions, which are grouped in a vector Lyapunov function through which the stability of the overall system is analyzed by taking into account the interconnection terms. This analysis
is performed either by constructing a scalar Lyapunov function on the basis of the components of the vector Lyapunov function [63, 43] or by using the comparison principle [115]. In the context of state-feedback of large interconnected systems, decentralization can be equivalently expressed as the computation of a state-feedback control law where the feedback gain matrix is block-diagonal. This structural constraint is of the same nature as the one associated to the output static feedback gain, which can be viewed as a full state feedback with structural constraints that select only the measured states [62]. In the state-feedback case [63], a block-diagonal gain is obtained by imposing a block-diagonal structure on a synthesis variable and the Lyapunov matrix. To remove the strong constraint on the Lyapunov matrix, the extended controller parameterization was developed in [43] to let the Lyapunov matrix remain unconstrained. It is to be noticed that the Lyapunov type approaches were generally associated with quite workable numerical algorithms, providing, at least for linear time invariant systems a practical and efficient method for decentralized control design.

1.2.2 Cooperative Control of Multi-agent Systems

Control of dynamic agents coupled to each other through an information flow network has emerged as a topic of major interest in recent years. In the following, we give an overview of the field organized by problems under study, compared to a categorization in Sec. 1.2.1 based on different methodologies. In addition, we establish a connection between consensus/formation control problem and nonhomogeneous matrix products.

Consensus Problem

In the past, a number of researchers have worked on problems that are closely related to consensus problems. Distributed agreement problems in Computer Science
have a long history [109]. The problem of reaching a consensus in a group of decision makers was studied in distributed estimation using a Bayesian framework [23, 181, 185]. Coordinated behaviors in nature, such as flocking/swarming and coupled oscillators, have been studied in ecology and biology, as well as in statistical physics and nonlinear science [89, 139, 173, 183, 190]. Engineering applications such as formation control have further increased the interest of engineers in swarming behaviors and collective motion patterns [55, 60, 80, 93, 101]. Very recent results on consensus problems include [11, 42, 73, 80, 95, 101, 120, 122, 114, 131, 148, 159, 169, 200], to name a few. These results can be broadly divided into several categories according to the main mathematical tools used: algebraic graph theory (e.g., [80, 131, 148]), nonlinear dynamics [120, 122], convex or iterative optimization [42, 200, 114], and partial contraction theory [169, 201].

In the following, we focus on the work most relevant to the present research and then point out the relationships among them. The work in [80] focuses on attitude alignment on undirected graphs. It is shown that the consensus on the heading angles of the agents can be achieved if the union of the interaction graphs for the team are connected frequently enough as the system evolves. Average-consensus problem is solved in [131] over strongly connected and balanced digraphs. Ren et al. [148] extend the results of [80] from the bidirectional case to unidirectional case. If the union of the collection of interaction graphs across some time interval had a spanning tree frequently enough, consensus (not necessarily average-consensus) can still be achieved. In [123], tools from nonlinear dynamics are used to obtain a broad class of communication patterns that guarantee global consensus. Xiao et al. propose a distributed iterative scheme, based on distributed average consensus in the network, to compute the maximum-likelihood estimate of the parameters [201]. This scheme works provided that the infinitely occurring communication
graphs are jointly connected. Other recent results on consensus problems include [11, 10, 42, 73, 95, 101, 114, 159, 169, 200], to name a few. See also [150] for a review on consensus problems in multi-agent coordination.

The aforementioned consensus protocols all operate in a synchronized fashion since each agent’s decisions must be synchronized to a common clock shared by all other agents in the group. This synchronization requirement might not be natural in certain contexts. For example, the agreement of time-on-target in cooperative attack among a group of UAVs depends in turn on the timing of when to exchange and update the local information. This difficulty entails the consideration of the asynchronous consensus problem, where each agent updates on its own pace, and uses the most recently received (but possibly outdated) information from other agents. No assumption is made about the relative speeds and phases of different clocks. Agents communicate solely by sending messages; however there is no guarantee on the time of delivery or even of a successful delivery. Therefore, heterogeneous agents, time-varying communication delays and packet dropout can all be taken into account in the same framework. Nevertheless, the asynchronism can destroy convergence properties that the algorithm may possess when executed synchronously or sequentially. Thus, the analysis of asynchronous algorithms is more difficult than that of their synchronous counterparts. We refer readers to [14, 57, 88] for surveys on general theory of asynchronous systems.

Work reported on asynchronous consensus problems is relatively sparse compared to its synchronous counterparts. A distributed iterative procedure under a weak form of asynchronism (called virtual synchronization) is developed by Mehyar et al for calculating averages over an unstructured peer-to-peer network [114]. Recently, we introduced an asynchronous framework to study the consensus problems for discrete-time multi-agent systems [53]. By combining linear asynchronous,
graph and matrix theories with a decomposition scheme, we prove, under a fixed information topology, the discrete-time asynchronous protocol achieves consensus asymptotically if the information exchange topology has a spanning tree. The most attractive aspect of this development is the fact that once the stability result was established for asynchronous protocols, the stability for synchronous protocols under dynamically changing interaction topologies is immediate since it can be seen as a special case of the asynchronous protocol with zero communication delays. Our asynchronous result extends the existing (synchronous) results reported in [80, 131, 148, 123, 114]. For other related problems in asynchronous multi-agent systems, see [12, 100, 104, 103, 105, 187].

Formation Control

Formation control has been a topic of significant interest to researchers in recent years. Of particular interest is the control of cooperative teams of vehicles to perform tasks such as reconnaissance missions and cooperative strikes. Note that formation control is, in a way, a consensus problem since in order to reach a stable formation the vehicles must achieve, among other things, the same velocity.

The primary distinction in the formation control literature is the type of formation control architecture used. An architecture may be broadly construed as a “coordination scheme,” and it determines the overall design approach for a specific formal control algorithm. In their survey [160], Scharf et al. divided formation flying control algorithms into five architectures: (i) Multiple-Input Multiple-Output (MIMO), in which the formation is treated as a single multiple-input, multiple-output plant, (ii) Leader/Follower (LF), in which individual spacecraft controllers are connected hierarchically, (iii) Virtual Structure (VS), in which spacecraft are treated as rigid bodies embedded in an overall virtual structure, (iv) Cyclic, in which individual
spacecraft controllers are connected non-hierarchically, and (v) Behavioral, in which multiple controllers for achieving different (and possibly competing) objectives are combined. The advantages and disadvantages of the various formation control architectures are discussed in [160]. In terms of “information requirements,” MIMO algorithms typically have the highest requirements, L/F moderate, and cyclic least.

Studies on how information structure affects the stability of formation have been made. String stability [176, 39] is concerned with a simple form of linear structure, and concepts in this area have been extended to a more general form of grid-like structure in the study of mesh stability [135]. More general information structures have also been considered. In their formulation of [55], the graph Laplacian of the information structure is connected to the stability of the formation and a Nyquist-like stability criterion is stated. A formal notion of formation stability in the context of graph rigidity is explored in [133]. Input-to-state stability (ISS) for formations is introduced in [180] and sufficient results, in terms of ISS of the individual agents, are given for stability of interconnections which are represented by tree-like graph structures. Stability analysis of interconnected systems where the topology might potentially be time varying are presented in [69, 65].

The sophistication of the control task is also dependent on the quality of the communication. Ketema and Balas investigate the effects of the rate of update of information among the vehicles in a formation and imprecise communications on the stability of the formation [87]. In follow-up work [86], they concentrate on the interaction between the continuous time dynamics of the agents and the discrete time nature of the information flow. In contrast, the results in [55] are based on a discrete model for the dynamics of the vehicles as well as for the flow of information. Different types of inaccuracy of communication (e.g. quantization error, ambient disturbances etc.) are modeled as additive white noises in [204]. Each vehicle uses
the covariance of the estimate of others’ positions as the measure of its available information quality.

Most of the work so far has centered on stability analysis of the formation assuming certain control laws in place. A more general question is that of synthesis of the control law for the agents, possibly, to optimize some performance metric. The defining feature of the problem is that the pre-specified topology of the formation imposes constraints on the form of the control law by limiting the information available to various agents at any time. Refer to Sec. 1.2.1 for a survey of optimal decentralized control methods subject to information constraints. With formation control application in mind, Gupta et al. study the synthesis of an LQR controller when the matrix describing the control law is constrained to lie in a particular vector space in [68, 70]. Based on the double-graph model, tools are developed to design and analyze the LF formation control strategies [83]. The feedback scheme proposed in [55] has been used and generalized by Lafferriere and his co-workers [90, 188, 194]. In particular, they prove that a necessary and sufficient condition for an appropriate decentralized linear stabilizing feedback to exist is the pre-specified communication digraph has a rooted directed spanning tree [90]. Receding horizon control offers an alternative approach to cooperative control design of MASs [50, 56, 204]. To the best of our knowledge, however, all the above work on formation control synthesis is developed under the assumption of a fixed communication graph.

Nonhomogeneous Matrix Products and Switched Systems

A matrix product $A^k$ is called homogeneous since only one matrix occurs as a factor. More generally, a matrix product $A_k \cdots A_1$ or $A_1 \cdots A_k$ is called a nonhomogeneous matrix product. Such products arise in areas such as nonhomogeneous Markov chains, demographics, probabilistic automata, production and manpower
systems, tomography, fractals, and designing curves. The book by Hartfiel [72] puts together much of the basic work on nonhomogeneous matrix products. A switched (linear) system is a dynamical system that consists of a finite number of (linear) subsystems described by differential or difference equations and a logical rule that orchestrates switching between these subsystems. It can be shown that a discrete time switched linear system is stable if and only if an infinite product of matrices taken from the set composed of subsystems’ $A$-matrices is convergent, i.e.,
\[
\lim_{k \to \infty} \| A_{i_k} \cdots A_{i_1} \| = 0.
\]
Hence, switched systems are technically related to nonhomogeneous matrix products. At the first glance, matrix products and switched systems appear to be out of place here. But they are not.

From the modeling point of view, a intriguing mathematical model for the setups discussed above in Sec. 1.2.2 is a switched interconnected system by the nature of the time-varying interaction among agents. However, the switching effects on the collective system behavior is not well understood due to the lack of suitable analysis tools. Recently, Jadbabaie et al. [80] provide a link between the consensus problem in a linear discrete time setup with undirected interaction graph and the convergence of infinite products of certain types of nonnegative matrices. Motivated by their work, there are a number papers that explore this connection to study the consensus or formation problems in different settings [131, 101, 148, 53, 201].

1.3 Organization of the Dissertation

The dissertation is organized as follows.

In Chapter 2, we study the consensus problem of multi-agent systems in the asynchronous framework. Under certain assumptions, the consensus protocol leads to stable behaviors even if the updating instants and sets of the agents are asynchronously determined. In particular, synchronous protocols under dynamically
changing interaction topologies can be seen as a special case of the asynchronous protocol where all communication delays are zero.

The goal of Chapter 3 is two fold. First, we explore communication topologies, as implied by the communication assumptions, that lead to consensus among agents. For this, several important results in the literature are examined and the focus is on different classes of communication assumptions being made, such as synchronism, connectivity, and direction of communication. In the latter part of this Chapter, we show that the confluent iteration graph unifies various communication assumptions and proves to be fundamental in understanding the convergence of consensus processes. In particular, based on asynchronous iteration methods for nonlinear paracontractions, we establish a new result which shows that consensus is reachable under directional, time-varying and asynchronous topologies with nonlinear protocols. This result extends the existing ones in the literature and have many potential applications.

Chapter 4 studies formation tracking problems which can be stated as multiple vehicles being required to follow pre-specified spatial trajectories while keeping a desired inter-vehicle formation pattern in time. We consider vehicles with nonlinear dynamics and nonholonomic constraints and very general trajectories that are generated by reference vehicles. We specify formations using the vectors of relative positions of neighboring vehicles and use consensus-based controllers in the context of decentralized formation tracking control. The key idea is to combine consensus-based controllers with the cascaded approach to tracking control, resulting in a group of linearly coupled dynamical systems. We examine the stability properties of the closed loop system using cascaded systems theory and nonlinear synchronization theory. Input-to-state stability (ISS) for formations is also introduced, and sufficient results, in terms of ISS of the individual agents, are given for stability
of interconnections which are represented by tree-like graph structures. Simulation results are presented to illustrate the proposed method.

Chapter 5 is dedicated to switched systems since besides being of relevance to the consensus/formation control problem switched systems are of theoretical interest on their own. It is a well-known fact that the asymptotic stability of switched linear systems under arbitrary switching sequences is equivalent to the robust stability of polytopic uncertain linear time-varying systems. By using tools from convex analysis, we first provide a new proof to this fact, which reveals a simple and clear link between two research fields. Some simple sufficient conditions for asymptotic stability of switching systems are given which become necessary and sufficient for several special cases. In the second part of this chapter, we investigate stability and control design problems with performance analysis for discrete-time switched linear systems. The switched Lyapunov function method is combined with Finsler’s Lemma to generate various tests in the enlarged space containing both the state and its time difference, allowing extra degree of freedom for stability analysis and control design. Two performance measures being considered are the decay rate and the input-output performance. Several LMI-based conditions are obtained to guarantee the stability and performance of the switched systems.
CHAPTER 2

INFORMATION CONSENSUS OF ASYNCHRONOUS DISCRETE-TIME
MULTI-AGENT SYSTEMS

2.1 Introduction

Information consensus is a fundamental problem in cooperative control that affects solutions to vehicle formation problems, flocking problems, rendezvous problems, attitude alignment, coordinated distributed decision making etc [11, 80, 101, 131, 148].

For cooperative control strategies to be effective, a team of agents should be able to respond to unanticipated situations or changes in the environment as a cooperative task is carried out. As the environment changes, the agents on the team must be in agreement as to what changes took place. A direct consequence of the fact that shared information is a necessary condition for coordination is that cooperation requires that the group of agents reaches consensus on the coordination data. The challenge here is for the group to have a consistent view of the coordination variables in the presence of unreliable, dynamically changing communication topology without global information exchange.

As mentioned in Section 1.2, most existing consensus protocols cannot be regarded as truly distributed because each agent’s decisions must be synchronized to a common clock shared by all other agents in the group. This synchronization requirement could be unrealistic. Consider a multi-robot system where robots com-
municate via wireless links. It is impossible for all robots to update their states at the same time due to channel contentions since all of them try to get information from neighbors. Motivated by this, we study the consensus problem for asynchronous discrete-time multi-agent systems. In such systems each agent operates according to its own clock; no assumption is made about the relative speeds of different clocks. Agents communicate solely by sending messages, however there is no guarantee on the time of delivery or even of a successful delivery. This scenario is actually prevalent in real systems such as distributed networks. We refer readers to [14, 57, 88] for surveys on the theory of asynchronous systems.

The asynchronism can destroy convergence properties that the algorithm may possess when executed synchronously or sequentially. Thus, the analysis of asynchronous algorithms is considerably more difficult than of their synchronous counterparts. Nevertheless, asynchronous systems give more reasonable models of the multi-agent systems in practical situations. For instance, heterogeneous agents, time-varying communication delays and packet dropout can be taken into account without much difficulty using the results from asynchronous theory. Also, in some situations asynchronous systems possess more robust properties than synchronous ones.

In this chapter, we propose an asynchronous framework to study the consensus problem. To facilitate the analysis, we start with the multi-agent systems under fixed interaction topologies, which have a rooted directed spanning tree. The spanning tree requirement is a milder condition than connectedness and is therefore more suitable for practical applications [148, 31]. Under certain assumptions, even if the updating instants of the agents are asynchronously determined, the consensus protocol continues to maintain the stability property. Despite its simple fixed topology, the model of asynchronous multi-agent systems encompasses those synchronous
ones with various communication patterns, i.e., we can address issues of directional, delayed, or time-variant communication in the same framework.

This work is not the first one that uses the asynchronous models to analyze multi-agent systems; see, e.g., [12, 100, 104, 103, 105]. Another related work is presented in [187], where decentralized asynchronous control and optimization schemes for stochastic discrete-event systems are analyzed. In the present work, synergistic union of graph theory, matrix theory and asynchronous theory allows us to reach very general results on the consensus problem.

The contributions of this chapter are two-fold. First, we extend the existing (synchronous) consensus results to the asynchronous setting. Under certain mild conditions, consensus can still be achieved for asynchronous multi-agent systems. Our asynchronous results shed new light on the existing (synchronous) results reported in [80, 131, 148, 120, 114]. Second, connections between synchronous and asynchronous protocols are established. As we show in this chapter, the asynchronous protocol under fixed topologies contain synchronous protocols under time-varying topologies as a special case. Then many of the techniques and results obtained earlier for asynchronous systems can be applied to study synchronous protocols, thus providing us with deeper understanding of the synchronous consensus dynamics. For example, the robustness of the synchronous protocol under time-varying topologies can be examined using the asynchronous theory.

This chapter is organized as follows. Section 2.2 describes the synchronous and asynchronous consensus problems considered in this work. In the asynchronous systems the order in which the states of agents are updated is not fixed and the selection of previous values of the states used in the updates is also not fixed. The main results are stated in Section 2.3, where we provide a framework for studying the asynchronous consensus problem by using asynchronous theory. Under certain
mild conditions, consensus can be achieved for asynchronous multi-agent systems.
In Section 2.4, the connections between synchronous and asynchronous protocols are established.

2.2 Preliminaries and Background

2.2.1 Definitions and Notations

Let $G = \{V, E, A\}$ be a weighted digraph (or direct graph) of order $n$ with the set of nodes $V = \{v_1, v_2, \ldots, v_n\}$, set of edges $E \subseteq V \times V$, and a weighted adjacency matrix $A = [a_{ij}]$ with nonnegative adjacency elements $a_{ij}$. The node indices belong to a finite index set $I = \{1, 2, \ldots, n\}$. A directed edge of $G$ is denoted by $e_{ij} = (v_i, v_j)$. For a digraph, $e_{ij} \in E$ does not imply $e_{ji} \in E$. The adjacency elements associated with the edges of the graph are positive, i.e., $a_{ij} > 0$ if and only if $e_{ji} \in E$. Moreover, we assume $a_{ii} \neq 0$ for all $i \in I$. The set of neighbors of node $v_i$ is the set of all nodes which point (communicate) to $v_i$, denoted by $N_i = \{v_j \in V : (v_j, v_i) \in E\}$.

A digraph $G$ can be used to model the interaction topology among a group of agents, where every graph node corresponds to an agent and a directed edge $e_{ij}$ represents a unidirectional information exchange link from $v_i$ to $v_j$, that is, agent $j$ can receive information from agent $i$. The interaction graph represents the communication pattern at certain time. The interaction graph is time-dependent since the information flow among agents may be dynamically changing. Let $\bar{G} = \{G_1, G_2, \ldots, G_M\}$ denote the set of all possible interaction graphs defined for a group of agents. Note that the cardinality of $\bar{G}$ is finite. The union of a collection of graphs $\{G_{i_1}, G_{i_2}, \ldots, G_{i_m}\}$, each with vertex set $V$, is a graph $\bar{G}$ with vertex set $V$ and edge set equal to the union of the edge sets of $G_{i_j}, j = 1, \ldots, m$.

A directed path in graph $G$ is a sequence of edges $e_{i_1i_2}, e_{i_2i_3}, e_{i_3i_4}, \ldots$ in that
Graph $G$ is called strongly connected if there is a directed path from $v_i$ to $v_j$ and from $v_j$ to $v_i$ between any pair of distinct vertices $v_i$ and $v_j$. Vertex $v_i$ is said to be linked to vertex $v_j$ across a time interval if there exists a directed path from $v_i$ to $v_j$ in the union of interaction graphs in that interval. A directed tree is a directed graph where every node except the root has exactly one parent. A spanning tree of a directed graph is a tree formed by graph edges that connect all the vertices of the graph.

Let $x_i \in \mathbb{R}$, $i \in \mathcal{I}$ represent the state associated with agent $i$. A group of agents is said to achieve global consensus asymptotically if for any $x_i(0)$, $i \in \mathcal{I}$, $\|x_i(t) - x_j(t)\| \to 0$ as $t \to \infty$ for each $(i, j) \in \mathcal{I}$. Besides being of interest in its own right, if consensus is attainable (all agents converge to a common point), then other formations are achievable too [101]. So the focus is on convergence to a point.

Let $\mathbf{1}$ denote an $n \times 1$ column vector with all entries equal to 1. Let $M_n(\mathbb{R})$ represent the set of all $n \times n$ real matrices. A matrix $F \in M_n(\mathbb{R})$ is nonnegative, $F \geq 0$, if all its entries are nonnegative, and it is irreducible if and only if $(I + |F|)^{n-1} > 0$, where $|F|$ denote the matrix obtained from $F$ by taking the absolute value of all entries. Furthermore, if all its row sums are +1, $F$ is said to be a (row) stochastic matrix.

### 2.2.2 Synchronous and Asynchronous Consensus Protocols

We consider the following (synchronous) discrete-time consensus protocol [148, 131]

$$x_i(t + 1) = \frac{1}{\sum_{j=1}^{n} A_{ij}(t)} \sum_{j=1}^{n} A_{ij}(t) x_j(t)$$

(2.1)

where $t \in \{0, 1, 2, \cdots\}$ is the discrete-time index, $(i, j) \in \mathcal{I}$ and $A_{ij}(t) > 0$ if information flows from $v_j$ to $v_i$ at time $t$ and zero otherwise, $\forall j \neq i$. The magnitude of $A_{ij}(t)$ represents possibly time-varying relative weighting or confidence of agent $i$. 

24
in the information state of agent $j$ at time $t$ or the relative reliabilities of information exchange links between them.

**Remark 2.1** If $A_{ij}(t)$’s are restricted to be 0 or 1, we call (2.1) the (unweighted) nearest neighbor rule, which was first introduced to control community in [80]. There are other ways to choose weights [201]: Maximum-degree weights. Here we use the constant weight $1/n$ on all edges, and choose the self-weights so that the sum of weights at each node is 1:

$$A_{ij}(t) = \begin{cases} 
\frac{1}{n}, & \text{if } \{i, j\} \in E(t) \\
1 - \frac{d_i(t)}{n}, & \text{if } i = j \\
0, & \text{otherwise}
\end{cases}$$

(2.2)

where $d_i$ is the out-degree of node $i$.

**Metropolis weights.** The Metropolis weight matrix is defined as

$$A_{ij}(t) = \begin{cases} 
\frac{1}{1+\max\{d_i(t), d_j(t)\}}, & \text{if } \{i, j\} \in E \\
1 - \sum_{\{i,k\} \in E(t)} A_{ik}(t), & \text{if } i = j \\
0, & \text{otherwise}
\end{cases}$$

(2.3)

With Metropolis weights, the weight on each edge is one over one plus the larger degree at its two incident vertices, and the self-weights are chosen so the sum of weights at each node is 1.

We can rewrite (2.1) in a compact form

$$x(t+1) = F(t)x(t)$$

(2.4)

where $x = [x_1, \cdots, x_n]^T$, $F = F_{ij}$ with $F_{ij} = \frac{A_{ij}(t)}{\sum_{j=1}^{n} A_{ij}(t)}$, $(i, j) \in I$. An immediate observation is that the matrix $F$ is a nonnegative stochastic matrix, which has an eigenvalue at 1 with the corresponding eigenvalue vector equal to 1.

The protocol (2.1) or (2.4) is localized since the state of agent $v_i$ only depends on the states of itself and its neighbors. Furthermore, this protocol is synchronous.
in the sense that all the agents update their states at the same time using the latest values of the states. From a practical point of view, since a central synchronizing clock may not exist and communication links create and fail dynamically, it is of interest to consider asynchronous multi-agent systems. In the asynchronous setting the order in which states of agents are updated is not fixed and the selection of previous values of the states used in the updates is also not fixed.

Now let \( t_0 < t_1 < \cdots < t_n < \cdots \) be the time instants when the state of the multi-agent system undergoes change. Let \( x_i(k) \) denote the state of agent \( i \) at time \( t_k \). The index \( k \) is also called the event-based discrete time index in the literature. The dynamics of asynchronous systems can be written as

\[
x_i(k + 1) = \begin{cases} 
\sum_{j=1}^{n} F_{ij}(k) x_j(k - d(i, j, k)) & \text{if } i \in S(k), \\
x_i(k) & \text{otherwise},
\end{cases}
\]

where \( d(i, j, k) \geq 0 \) are nonnegative integers, \( S(k) \) are nonempty subsets of \( \{1, \cdots, n\} \), the initial states are specified by \( x(0) = x(-1) = \cdots \) Henceforth, we write the initial vector \( x(0) \) to abbreviate reference to this set of equal initial states. We refer to the \( d(i, j, k) \) as iteration delays and \( S(k) \) as updating sets.

**Remark 2.2** Numerous asynchronous models have been proposed and successfully applied to some practical problems [14, 57, 88]. Examples of asynchronous systems are multiprogramming computer systems, concurrent/distributed database systems, process-control systems such as a factory automation system or a fly-by-wire aircraft controller, and communication protocols. Eq. (2.5) is one of the prevalent models in asynchronous theory [174].

The interpretation of (2.5) in the modeling of consensus protocols of multi-agent systems is as follows. Consider the following scenario. A team of five mobile robots navigate in a terrain with obstacles while maintaining a desired formation
as illustrated in Fig. 2.1(a). The robots are heterogeneous. There are three types of robots (gray, gridded and white) and every type has its own clock, which implies that the updating moments for different agents may not coincide. For example, at time (or more precisely, iteration) $k_1$, only gridded agents 2 and 4 update their states while other agents keep their states unchanged, i.e., $S(k_1) = \{2, 4\} \subseteq \{1, 2, \cdots , 5\}$.

Moreover, the interaction graph dynamically changes during the time of passing the obstacle (Figs. 2.1(b), (c), and (d)). At iteration $k_2$, Agent 1 may calculate its new state $x_1(k_2 + 1)$ using the old information from agent 3. A snapshot of the system (Fig. 2.1(b)) shows that $x_1(k_2 + 1) = 1/4 [x_1(k_2) + x_2(k_2) + x_3(k_2 - 1) + x_4(k_2)]$ or $d(1,3,k_2) = 1$. It is not difficult to see that the synchronous model (2.1) is a special case of (2.5) with no delays and the updating sets $S(t) = \{1, \cdots, n\}$ for all $t$.

**Remark 2.3** The delays $d(i,j,k)$ in (2.5) are time-varying and link-dependent. Consensus problems with time-delays have been considered by previous research, e.g., in [122, 131], but the delays considered were constant for all communication links. Sometimes, we can purposely stagger the data communication between agents and thereby reduce the associated delays.
2.2.3 General Stability Theorems for Asynchronous Systems

In this section, we focus on linear asynchronous systems satisfying the regularity assumption:

- There exists a nonnegative integer $D$ such that
  \[ 0 \leq d(i, j, k) \leq D < \infty, \quad \forall (i, j, k). \tag{2.6} \]

Condition (2.6) indicates that only a finite number of updating instants can occur within any time interval of finite length. In the literature, this is also called partially asynchronism or (uniformly) bounded-delay asynchronism.

- The updating sets $S(k)$ satisfy
  \[ \bigcup_{k=K}^{\infty} S(k) = \{1, \cdots, n\}, \text{ for any } K. \tag{2.7} \]

Condition (2.7) says that every agent is updated infinitely often. In other words, no agent fails to be updated as time goes on.

The following results due to Chazan and Miranker [32], Su et al. [174], and Lubachevsky and Mitra [106] are fundamental in the theory of stability of asynchronous systems.

**Lemma 2.1** Let $F(k) = [F_{ij}(k)] = F$, $\forall k$. The fixed point of (2.5) is asymptotically stable under the class of regular asynchronisms if the spectral radius of $|F|$, $\rho(|F|)$, is less than unity. Here the absolute operation is understood to be element-wise.

**Remark 2.4** When the context is clear, the absolute operation $|\cdot|$ applied on $F$ is dropped hereafter since it is always nonnegative in our setting.

It is worth mentioning that $\rho(|F|) < 1$ is a worst-case condition since it has been formulated in terms of ‘all possible asynchronous systems with a given matrix,’ but not in terms of an individual system with a given updating law. Nevertheless, the
condition is tight; if $\rho(|F|) > 1$, and if 1 is not an eigenvalue of $F$, an initial vector $x(0)$ and a sequence $d(i,j,k) \leq 1$, $i,j \in I$, $k \in \mathbb{N}$, can be constructed for which (2.5) does not converge [174].

A commonly, but mistakenly, held belief is that the condition $\rho(|F|) < 1$ is also necessary for the convergence of asynchronous iterations [32, 14, 174]. The source of this error was identified only recently by Szyld in [178]. If $\rho(|F|) = 1$, under certain conditions, convergence can still be achieved. Lubachevsky and Mitra [106], and more recently Pott [142], studied asynchronous methods with singular matrices, i.e., for the specific case of $\rho(F) = 1$ and gave conditions for the convergence of the asynchronous iteration (2.5). Additional assumptions to (2.6) and (2.7) made in [106] are as follows. For at least one agent, say $i$, $i \in I$,

$$F_{ii} > 0, \quad (2.8)$$

$$d(i,i,k) = 0, \quad (2.9)$$

$$x_i(0) > 0. \quad (2.10)$$

**Lemma 2.2 ([106])** Consider the asynchronous system (2.5) with a nonnegative irreducible matrix $F$ of spectral radius unity. Assume (2.6)-(2.10) and initial state $x(0)$ is nonnegative. Then, there exists a positive, finite constant $b$ such that, as $k \to \infty$,

$$x(k) \to b\pi,$$

where $\pi$ is a (column) vector satisfying $F\pi = \pi$. Furthermore, the constant $b$ is bounded (from below and above) and its value depends on $x(0)$, $F$, and the sequence of the update sets $S(k)$ and delays $d(i,j,k)$.
2.3 Asynchronous Information Consensus

In this section, the asynchronous information consensus problem is studied. We focus on the asynchronous consensus protocol under a (structurally) fixed topology which has a rooted directed spanning tree. A directed spanning tree only imposes a mild condition in the stability analysis of synchronous case as shown in [148, 31]. A sufficient condition is derived to guarantee the convergence of the asynchronous consensus protocol. Even under such seemingly ideal topologies, the asynchronous protocol turns out to be very powerful in studying the synchronous counterparts under a large class of interaction patterns.

2.3.1 Synchronous Case with Fixed Topologies

Let us first review the known synchronous consensus results with fixed interaction topologies. We present them here to allow for a comparison with the asynchronous model in the following sections.

**Lemma 2.3 ([148])** Let \( F \in M_n(\mathbb{R}) \) be a stochastic matrix with positive diagonal entries. The matrix \( F \) has a unique eigenvalue at 1 with maximum modulus if and only if the graph associated with \( F \) has a spanning tree. In this case, \( \lim_{m \to \infty} F^m = \mu^T \), where \( m \in \mathbb{N}^+ \) and \( \mu = [\mu_1, \ldots, \mu_n]^T \geq 0 \) satisfies \( \mu^T \mu = 1 \).

**Remark 2.5** The diagonal elements of \( F \) are positive since we assume that there is a link from each vertex to itself.

**Theorem 2.1 ([148])** Given the synchronous protocol (2.1) with \( F(k) = F, \forall k \in \mathbb{N} \), the consensus is asymptotically reachable if and only if the associated interaction graph \( G \) has a spanning tree. That is, global consensus is asymptotically reachable.

**Example 2.1** This example demonstrates that the consensus point depends on the initial states of the agents and the interaction topologies. Fig. 2.2 shows that our
Figure 2.2. The (synchronous) consensus value depends on the initial states and the interaction topologies.

Multi-agent systems seek consensus with the same initial states $x_a(0) = x_b(0) = x(0) = x_d(0) = [3 \ 5 \ 2 \ 1]^T$ but under different interaction topologies. All four topologies have a directed or undirected spanning tree with unit weighted links. The reader can verify that the consensus points are

\[
\begin{align*}
x_a(\infty) &= [3 \ 3 \ 3 \ 3]^T, \\
x_b(\infty) &= [2 \ 2 \ 2 \ 2]^T, \\
x_c(\infty) &= [3.2 \ 3.2 \ 3.2 \ 3.2]^T, \\
x_d(\infty) &= [2.9 \ 2.9 \ 2.9 \ 2.9]^T.
\end{align*}
\]

In particular, if no other communication links exist in the topology but the directed spanning tree, then all agent states converge to the initial state of the root agent. For example, all the agents in Fig. 2.2(a) eventually reach a value of 3, which is exactly the initial state of agent 1. Another very important observation is that every agent state is bounded from below and above,

\[
\min_{j=1}^n x_j(0) \leq x_i(k) \leq \max_{j=1}^n x_j(0), \ \forall i \in I.
\] (2.11)
This is because the dynamics of (2.1) are based on the process of taking averages.

2.3.2 Asynchronous Case with Fixed Topologies

As we shall demonstrate below, the convergence process of asynchronous protocol (2.5) is fundamentally different from that of the synchronous protocol (2.1).

In our case, to reach a consensus the matrix $F$ must have a spectral radius equal to one by the necessity part of Lemma 2.3. Hence, we are dealing with a singular ($\rho(F) = 1$) asynchronous protocol and Lemma 2.2, instead of Lemma 2.1, should be invoked to get a correct condition.

Before using Lemma 2.2, let us first check whether or not the assumptions (2.8)-(2.10) are satisfied for the asynchronous system (2.5). The assumption (2.8) is satisfied by Remark 2.5. In the multi-agent application, it is reasonable to assume that every agent always uses its latest state to calculate the new state, i.e., the condition $d(i, i, k) = 0$ is true for at least one agent. Finally, the assumption (2.10) may not be a practical restriction at all if the simple expedient of taking all initial states positive is followed. It is needed to guarantee the consensus points to be bounded away from the trivial solution 0.

Now we are ready to state the convergence result for the asynchronous protocol, which is based on Lemma 2.2 and Theorem 2.1 combined with a decomposition idea.

**Theorem 2.2** Consider the asynchronous protocol (2.5) with structurally fixed topology $F(k) = F$, $\forall k \in \mathbb{N}$. Assume that all the agents have access of their own states (i.e., $F$ has positive diagonal entries) and at least one of them can access its own state without delay. Then the consensus is asymptotically reachable if the associated (directed) graph $G$ has a spanning tree. That is, global consensus is asymptotically reachable under the asynchronous mode.
Figure 2.3. Asynchronous multi-agent systems with different interaction topologies.

Sketch of the Proof: In the following, we only provide a graphical explanation of the Theorem. A rigorous proof can be obtained by using Theorem 3.4.

In Lemma 2.2, the irreducibility of $F$ is equivalent to the condition of a strongly connected graph (Theorem 6.2.24 [78]). However, this is not necessarily true since the graph considered here only has a spanning tree. In fact, the irreducibility condition is only sufficient but not necessary in the proof of Lemma 2.2. It is used to index agent numbers in such a way so that

\[ F_{11} > 0 \text{ and } x_1(0) > 0, \quad (2.12) \]

\[ \sum_{i=1}^{j-1} F_{ij} > 0, \quad j = 2, 3, \ldots, n. \quad (2.13) \]

From previous discussions, we know that condition (2.12) is easily satisfied. It remains to show that (2.13) holds when agent numbers are appropriately indexed. Consider the following cases.

Case 1. The graph is strongly connected. Then $F$ is irreducible and (2.13) holds by Proposition 3 in [106].

Case 2. The graph has a linear structure as shown in Fig. 2.3(b), i.e., all the
agents can be visited once (and only once) along a directed path. Label the agent numbers sequentially in the reversed direction of the directed path, then every agent \(i\) \((0 \leq i \leq n-1)\) except for agent \(n\) has agent \((i+1)\) as its neighbor. The matrix \(F\) thus has all its superdiagonal elements greater than zero. Hence, (2.13) is satisfied since \(F\) is nonnegative.

From Lemma 2.2 and Theorem 2.1, global consensus is asymptotically reachable for asynchronous multi-agent systems in both Cases 1 and 2.

**Case 3.** To study the most general case, we use a decomposition idea to partition the interaction graph by a minimum subset of unidirectional links so that every resulting subgraph is either strongly connected or has a spanning tree.

Hereafter, a concrete example is considered but the analysis is applicable to any graph having a spanning tree structure. The graph \(G_{3c}\) in Fig 2.3(c) has a spanning tree rooted at node 1 with four unidirectional links \(e_{17}, e_{35}, e_{56},\) and \(e_{89}\). The graph can be partitioned by \(e_{17}\) and \(e_{35}\) into three subgraphs \(G_{s1}, G_{s2},\) and \(G_{s3}\) (i.e., \(G_{3c} = \{G_{s1}, G_{s2}, G_{s3}\} \cup \{e_{35}, e_{17}\}\)) such that they are either strongly connected \((G_{s1}\) and \(G_{s3}\)) or have a linear structure \((G_{s2})\). There is no need to relax link \(e_{56}\) \((e_{89})\) since \(G_{s2}\) \((G_{s3})\) already carries a linear structure (is strongly connected).

It is observed that the subgraph \(G_{s1}\), which contains the spanning tree root, is actually decoupled from subgraphs \(G_{s2}\) and \(G_{s3}\). That is, it only transmits state information to \(G_{s2}\) and \(G_{s3}\), but never receives information from them since the links \(e_{17}\) and \(e_{35}\) are unidirectional. From the analyses of Cases 1 and 2, all the nodes in \(G_{s1}\) can asymptotically reach consensus in an asynchronous mode, i.e., \(\forall \epsilon/2 > 0, \exists k_1 > 0, \) such that

\[
|x_i(k_1) - x_j(k_1)| < \epsilon/2, \; \forall i, j \in V_{s1}.\tag{2.14}
\]

Again, due to the directedness of \(e_{17}\) and \(e_{35}\), the state of every node in \(G_{s3}\) and \(G_{s2}\) eventually converges to the state of agent 1 and agent 3, respectively (cf. Example
2.1). Mathematically, \( \forall \epsilon/2 > 0, \exists k_2, k_3 > 0, \) such that
\[
|x_3(k_2) - x_j(k_2)| < \epsilon/2, \; \forall j \in V_{s2}, \tag{2.15}
\]
\[
|x_1(k_3) - x_j(k_3)| < \epsilon/2, \; \forall j \in V_{s3}. \tag{2.16}
\]

From (2.14), (2.15) and (2.16), we have: \( \forall \epsilon > 0, \exists k_m > \max\{k_1, k_2, k_3\}, \) such that
\[
|x_i(k_m) - x_j(k_m)| < \epsilon, \; \forall i, j \in V_{3c}. \tag{2.17}
\]

This means that all the agents in the graph \( G_{3c} \) will eventually reach consensus. \( \square \)

Although Theorems 2.1 and 2.2 look similar, they are fundamentally different. In the synchronous case, \( x(t) \) converges to a consensus point, which is only a function of the interaction topology and initial states and otherwise independent of the computations; see Example 2.1. In the latter case, \( x(k) \) converges to a consensus point depending on the computations, that is, on the update sets, the delays, and initial states. An intuitive explanation of why this is the case will be given in the next section. The only exception is when the interaction graph has a unique rooted spanning tree, states of all the nodes asymptotically converge to the initial state of the root node. For instance, considering the systems shown in Figs. 2.2(a) and (b), \( x_a(k) \to [3 3 3 3]^T \) and \( x_b(k) \to [2 2 2 2]^T \) as \( k \to \infty \) regardless of computations.

**Example 2.2** This example shows that the asynchronous consensus value generally depends on the course of the computations. A multi-agent system with five agents and a (structurally fixed and equally weighted) interaction topology is shown in Fig. 2.4. Since the interaction graph has a directed spanning tree, the consensus is reachable under asynchronous updating. With the initial condition \( x(0) = [-2 1 2 0 -4]^T \), the asynchronous consensus value of the system is studied. At every iteration, a node is chosen to update its state randomly and independently of other nodes with probability \( p \). The delay \( d(i, j, k) \) in (2.5) is a discrete
random variable taking an integer value between 0 and $D$ with an equal probability. All randomizations across the nodes and across the iterates are independent in the simulations.

Monte Carlo simulation experiments were conducted. Fig. 2.5 shows different consensus values for 2000 independent runs with node selection probability $p = 1/2$ and the delay bound $D = 0$ (zero-asynchronism). From this figure, it is clear that the asynchronous consensus value can take any value in a bounded range. The corresponding histogram (Fig. 2.6) is unimodal. Interestingly, the mean of the asynchronous consensus values is very close to the synchronous consensus value, i.e., -0.1176. Other experiments (with different interaction topologies and different initial conditions) also confirm the above findings.

The effects of the magnitudes of the node selection probability $p$ and the delay bound $D$ on the consensus dynamics are also explored. On one hand, the asynchronous systems tend to behave more like synchronous systems with increasing $p$ (for fixed topology $F$ and delay bound $D$). As expected, this effect is less noticeable when $D$ becomes large. On the other hand, the reader may expect that decreasing $D$ (with fixed $p$) has a similar effect on asynchronous systems as increasing $p$. 

Figure 2.4. The interaction topology of an asynchronous multi-agent system with five agents.
Figure 2.5. Asynchronous consensus value \((p = 1/2 \text{ and } D = 0)\) depends on various topology selections made in the course of the computations.

Figure 2.6. The histogram of the consensus value \((p = 1/2 \text{ and } D = 0)\) is unimodal.
However, this turns out to be false. The histogram of consensus value for 2000 simulation runs is shown in Fig. 2.7. It can be seen the consensus region instead shrinks by more than 40%. The reason behind this counterintuitive result is under investigation.

2.4 Synchronous Consensus with Time-varying Topologies: An Asynchronous Perspective

We have addressed the asynchronous consensus problem under a (structurally) fixed topology, which has the same directed spanning tree during the consensus process. This obviously is a quite restrictive requirement. However, we need to emphasize that the topology is only invariant in the “structure domain” or $F(k) = F$, $\forall k \in \mathbb{N}$ in (2.5). In the time domain, the interaction between agents may be time-dependent due to asynchronisms. Can we map the synchronous systems under time-varying interaction topology into a particular type of asynchronous systems? The answer is affirmative as shown below, thus results from asynchronous systems
can be used to study the synchronous consensus protocols. Theorem 2.2 forms the basis of the following development.

Let us consider a special type of asynchronous multi-agent systems under a fixed interaction topology. All the agents in the system update their states always at the same time instants (say, when the updating mechanism is triggered by a time impulse coming from an external clock). The asynchronism is solely caused by the loss and creation of communication links between different agents. In fact, this kind of asynchronous systems, with the updating set \( S(k) = \{1, \cdots, n\} \) for all \( k \) and non-zero delays, can be seen as synchronous systems under time-varying topologies. To see this, we give a simple example. In the sequel, we no longer differentiate the discrete time index \( t \) and event based discrete time index \( k \).

**Example 2.3** Consider an asynchronous multi-agent system under a (structurally) fixed topology, which has a directed spanning tree, \( G_{2a} \) as shown in Fig. 2.2(a). The updating sets satisfy \( S(t) = \{1, 2, 3\} \) for all \( t \) and time delays satisfy \( 0 \leq d(i, j, t) \leq D \) with \( D \) a finite nonnegative integer. Due to time delays, at a particular time instant the interaction topology may correspond to one of the graphs from Fig. 2.8(a).

On the other hand, a synchronous multi-agent system under time-varying topologies is shown in Fig. 2.8(b). At every time instant, the system has an interaction graph which is the superset of one of the graphs from Fig. 2.8(a), e.g., \( G(t+1) \supseteq G_{21} \). Assume that the time interval \( [t, t+T_1] \) is long enough such that \( \bigcup_{q=0}^{T_1} G(t+q) = G_{2a} \). Across this time interval, the synchronous system under time-varying topologies maps into the asynchronous system. Thus they have the same stability property in terms of consensus reaching. From the pigeon-hole principle, \( T_1 = DC(n) + 1 \) is enough to guarantee the union of the interaction graphs containing a spanning tree, where \( C(n) = 2^{n-1} \) (\( n = 4 \)) is the total number of graphs in Fig. 2.8(a).
Figure 2.8. Synchronous systems with time varying topologies can be seen as a special case of asynchronous systems.
In Example 2.3, the mapping between an asynchronous system and a synchronous system is established by assuming that the union of the synchronous interaction graphs across some time interval contains the same spanning tree as the asynchronous system. This assumption is not restrictive and can be relaxed. For a system with $n$ agents, the total number of different directed spanning tree configurations, denoted as $L(n)$, is finite. Again using the pigeon-hole principle, we can conclude that there must exist the same spanning tree across each time interval of length $T_2 = (L(n) + 1) \times T$, if the union of graphs across each time interval of length $T(\leq T_1)$ has a spanning tree (not necessarily the same one). Note that $T_2$ is a finite number since $L(n)$ and $T$ are both bounded. If the asynchronous system under the interaction topology with such a spanning tree can asymptotically reach consensus, so can the synchronous system as $t \to \infty$.

Based on the above discussions, we have the following result.

**Theorem 2.3** Let $G(t) \in \bar{G}$ be a time-varying interaction graph at time $t$, with the weights selected from a finite set of arbitrary positive numbers. The protocol (2.5) achieves global consensus asymptotically if and only if there exists an infinite sequence of contiguous, nonempty, bounded time intervals $[t_l, t_{l+1})$, $l \geq 0$, starting at $t_0 = 0$, with the property that across each such interval, the union of the interaction graphs has a spanning tree.

**Proof.** (Sufficiency.) Let $T$ denote the least upper bound on the lengths of the intervals $[t_l, t_{l+1})$, $l \geq 0$. By assumption, $T < \infty$. Let the cardinality of the weight set be $W$. The synchronous system can be mapped into an asynchronous system with times delays. Since there must exist the same spanning tree across every time interval of length $WT[L(n)+1]$, the bound of delays is given by $D = \left\lceil \frac{WT[L(n)+1]-1}{C(n)} \right\rceil$, where $n$ is the agent number in the system, $L(n)$ the total number of spanning tree configurations for a graph with $n$ vertices, and $C(n) = 2^{(n-1)}$. By Theorem 2.2, the
sufficiency follows.

(Necessity.) Straightforward and is omitted. □

Several comments are appropriate. First, Theorem 2.3 is in essence the same to Proposition 1 in [120] and Theorem 3.3 in [148] but the proof techniques are totally different. Here, a new approach is taken by combining asynchronous theory with graph theory and matrix theory. Second, the synchronous consensus point depends on the time-varying topologies. (A similar effect was shown in the distributed estimation systems that an estimate based on processed data depends upon the ordering of interprocessor messages [23].) From the asynchronous point of view, this is because the same synchronous system (under the time-varying topologies) may be mapped to different asynchronous systems with different spanning trees. The consensus point depends on the sequence of the spanning trees in the course of the computations; c.f. Example 2.1. This also gives a physical example why the constant $b$ in Lemma 2.2 is computationally dependent. Tight lower and upper bounds for the constant $b$ are given in [106]. It is also shown that a measure of the projective distance of $x(k)$ from the consensus point vanishes at least at a geometric rate.
CHAPTER 3

ON COMMUNICATION REQUIREMENTS FOR MULTI-AGENT CONSENSUS SEEKING AND NONLINEAR CONSENSUS PROTOCOLS

3.1 Introduction

Work reported on the asynchronous consensus problem is relatively sparse compared to its synchronous counterparts. In Chapter 2, we introduced an asynchronous framework to study the consensus problems for discrete-time multi-agent systems with a fixed communication topology under the spanning tree assumption (All the assumptions in this paragraph will be discussed in detail later). A distributed iterative procedure under the eventual update assumption was developed in [114] for calculating averages on asynchronous communication networks. The asynchronous consensus problem with zero time delay was studied in [29] where the union of the communication graphs is assumed to have a common root spanning tree. A nice overview of the asynchronous consensus problem is given in [21] where the authors link the consensus problem considered here to earlier work [186]. For other related problems in asynchronous multi-agent systems, see [12, 100, 105, 187].

Asynchronism provides a new dimension to consensus problems and makes convergence harder to achieve. Under certain technical conditions, asynchronism is not detrimental to consensus seeking among agents. A natural question is what are the appropriate requirements on communication topologies to guarantee the convergence of consensus processes? In order to answer this question, we first discuss the various
assumptions on communication topologies commonly used in the literature and classify several of the existing consensus results by these communication assumptions. In Sect. 3.3, we prove the convergence of asynchronous consensus with zero time delay involving pseudocontractive mappings. This development is a generalization of the results of [29] and [201], and provides insight into why the choice between bidirectional and unidirectional communication assumptions can make the difference in establishing consensus convergence. In Sect. 3.4, we unify various communication assumptions using the confluent iteration graph proposed in [142]. Furthermore, a new convergence result for nonlinear protocols is developed based on the confluent asynchronous iteration concept. This result contains some existing ones in [21] as special cases.

3.2 Preliminaries and Background

3.2.1 Definitions and Notations

Let $X^*$ be a nonempty closed convex subset of $\mathbb{R}^n$, and let $\| \cdot \|$ be a norm on $\mathbb{R}^n$. For any vector $x \in \mathbb{R}^n$, $y^* \in X^*$ is a projection vector of $x$ onto $X^*$ if $\|x - y^*\| = \min_{y \in X^*} \|x - y\|$. We use $P(x)$ to denote an arbitrary but fixed projection vector of $x$ and dist($x, X^*$) to denote $\|x - P(x)\|$. Let $T$ be an operator on $\mathbb{R}^n$. It is paracontractive if

$$\|Tx\| \leq \|x\| \quad \text{for all} \quad x \in \mathbb{R}^n$$

and equality holds if and only if $Tx = x$. An operator is nonexpansive (with respect to $\| \cdot \|$ and $X^*$) if

$$\|Tx - x^*\| \leq \|x - x^*\| \quad \text{for all} \quad x \in \mathbb{R}^n, x^* \in X^*,$$

and pseudocontractive [175] (with respect to $\| \cdot \|$ and $X^*$) if, in addition,

$$\text{dist}(Tx, X^*) < \text{dist}(x, X^*) \quad \text{for all} \quad x \notin X^*.$$
We use $\mathcal{T}$ to denote the set of all pseudocontractive operators. In the linear case, pseudocontractive operators are generalizations of paracontractive ones. But the converse is not true. Consider the following inequalities:

$$\|Tx - P(Tx)\| \leq \|Tx - P(x)\| \leq \|x - P(x)\|, \text{ for all } x \notin X^*.$$  \hfill (3.4)

Paracontractivity requires the second inequality to be strict, while pseudocontractivity requires any one of these two inequalities to be strict.

**Example 1.** Let $T \in \mathbb{R}^{n \times n}$, $X^* = \{c\mathbf{1}|c \in \mathbb{R}\}$. Then $T$ is pseudocontractive with respect to $X^*$ and the infinity norm $\|\cdot\|_\infty$ if and only if $T\mathbf{1} = \mathbf{1}$ and for any $x \in \mathbb{R}^n$ such that $\min_i x_i < \max_i x_i$, $\max_i (Tx)_i - \min_i (Tx)_i < \max_i x_i - \min_i x_i$. \hfill $\diamond$

### 3.2.2 Synchronous and Asynchronous Consensus Protocols

We consider the following (synchronous) discrete-time consensus protocol [149, 131, 120]

$$x_i(t+1) = \frac{1}{\sum_{j=1}^n a_{ij}(t)} \sum_{j=1}^n a_{ij}(t)x_j(t)$$  \hfill (3.5)

where $t \in \{0, 1, 2, \cdots\}$ is the discrete-time index, $(i, j) \in \mathcal{I}$ and $a_{ij}(t) > 0$ if information flows from $v_j$ to $v_i$ at time $t$ and zero otherwise, $\forall j \neq i$. The magnitude of $a_{ij}(t)$ possibly represents time-varying relative confidence of agent $i$ in the information state of agent $j$ at time $t$ or the relative reliabilities of information exchange links between agents $i$ and $j$. We can rewrite (3.5) in a compact form

$$x(t+1) = F(t)x(t)$$  \hfill (3.6)

where $x = [x_1, \cdots, x_n]^T$, $F(t) = [F_{ij}(t)]$ with $F_{ij}(t) = \frac{a_{ij}(t)}{\sum_{j=1}^n a_{ij}(t)}$, $(i, j) \in \mathcal{I}$. An immediate observation is that the matrix $F$ is a nonnegative stochastic matrix, which has an eigenvalue at 1 with the corresponding eigenvector equal to $\mathbf{1}$. The protocol (3.5) or (3.6) is synchronous in the sense that all the agents update their states at the same time using the latest values of neighbors’ states.
The way described above to define $F(t)$ in (3.6) is only one possible way among many others. In the following we assume that $F(t)$ satisfies Assumption 1 below.

**Assumption 3.1 (Nontrivial interaction strength [21])** There exists a positive constant $\alpha$ such that

(a) $F_{ii}(t) \geq \alpha$, for all $i, t$.

(b) $F_{ij}(t) \in \{0\} \cup [\alpha, 1]$, for all $i, j, t$.

(c) $\sum_{j=1}^{n} F_{ij}(t) = 1$, for all $i, t$.

Now, in the asynchronous setting the order in which states of agents are updated is not fixed and the selection of previous values of the states used in the updates is also not fixed. Now let $t_0 < t_1 < \cdots < t_n < \cdots$ be the time instants when the state of the multi-agent system undergoes change. Let $x_i(k)$ denote the state of agent $i$ at time $t_k$. The index $k$ is also called in the literature the *event-based discrete time index*. The dynamics of asynchronous systems can be written as

$$x_i(k+1) = \begin{cases} \sum_{j=1}^{n} F_{ij}(k)x_j(s_j^i(k)) & \text{if } i \in I(k), \\ x_i(k) & \text{if } i \notin I(k), \end{cases}$$  

(3.7)

where $s_j^i(k)$ are nonnegative integers, $I(k)$ are nonempty subsets of $\{1, \cdots, n\}$. The initial states are specified by $x(0) = x(−1) = \cdots$. Henceforth, we write the initial vector $x(0)$ to abbreviate reference to this set of equal initial states. We refer to $d_j^i(k) = k - s_j^i(k)$ as *iteration delays* and $I(k)$ as *updating sets*. The following (*partial asynchronism/regularity*) assumptions are usually made in the study of linear asynchronous linear systems.

**Assumption 3.2 (Partial asynchronism)** (a) (Frequency of updating) The updating sets $I(k)$ satisfy

$$\exists B_1 \geq 0, \bigcup_{k=i}^{i+B_1} I(k) = \{1, \cdots, n\}, \text{ for all } i.$$  

(3.8)
(b) (Bounded-delay asynchronism) There exists a nonnegative integer $B$ such that
\[ 0 \leq k - s^i_j(k) \leq B_2 < \infty, \forall (i, j, k). \] (3.9)

(c) $s^i_i(k) = k$, for all $i$.

Assumption 3.2(a) says that every agent should be updated at least once in any $B + 1$ iteration steps. This condition is also known as a regulated condition in the literature. Assumption 3.2(b) requires delays to be bounded by some constant $B$. Assumption 3.2(c) says that an agent generally has access to its own most recent value. Without loss of generality (but after renumbering in the original definition), we assume that $I(k)$ is a singleton which contains a single element from $\{1, \ldots, n\}$.

Furthermore, if for all $i$ there exists a nonnegative integer $B_i$ such that for all $j$
\[ i \in \bigcup_{k=j}^{j+B_i} I(k), \] (3.10)
we call $I(k)$ an indexwise-regulated sequence [142]. This condition expresses the fact that different agents may have different updating frequencies.

3.2.3 Other Communication Assumptions

Let us review several important assumptions commonly used in the literature.

Assumption 3.3 (Connectivity)  (a) (Uniform strong connectivity) There exists a nonnegative integer $B$ such that $\bigcup_{s=t}^{t+B} G(s)$ is strongly connected for all $t$.

(a') (Uniform spanning tree) There exists a nonnegative integer $B$ such that $\bigcup_{s=t}^{t+B} G(s)$ contains a spanning tree for all $t$.

(b) (Nonuniform strong connectivity) $\bigcup_{t \geq t_0} G(t)$ is strongly connected for all $t_0 \geq 0$.

(b') (Nonuniform spanning tree) $\bigcup_{t \geq t_0} G(t)$ contains a spanning tree for all $t_0 \geq 0$.

Assumption 3.4 (Direction of Communication)  (a) (Bidirectional link) If $e_{ij} \in G(t)$ then $e_{ji} \in G(t)$. It implies that the updating matrix $F(t)$ or $F(k)$ is symmetric.

(b) (Unidirectional link) $e_{ij} \in G(t)$ does not imply $e_{ji} \in G(t)$. In this case, the updating matrix $F(t)$ or $F(k)$ is not necessarily symmetric.
Assumption 3.5 (Reversal link) 

(a) If \((v_i, v_j) \in G(t)\), then there exists some \(\tau\) such that \(|t - \tau| < B\) and \((v_j, v_i) \in G(\tau)\) [29, 21].

(b) There is a nonnegative integer \(B\) such that for all \(t\) and all \(v_i, v_j \in V\) we have that if \((v_i, v_j) \in G(t)\) then \(v_j\) is linked to \(v_i\) across \([t, t + B]\) [121].

Assumptions 3.1-3.5 play different roles in proving various consensus results and they are not necessarily independent from each other. Assumption 3.1 (Nontrivial interaction strength) and one of the items in Assumption 3.3 (Connectivity) are always necessary for the convergence of consensus protocols. They guarantee that any update by any agent has a lasting effect on the states of all other agents. Assumption 3.2 (Partial asynchronism) describes a class of asynchronous systems.

If Assumption 3.3(a) (Uniform strong connectivity) is satisfied, then Assumption 3.2(a) (Frequency of Updating) and Assumption 3.5(b) (Reversal link) are satisfied automatically. Assumption 3.4(a) (Bidirectional link) is a special case of Assumption 3.5 (Reversal link). Instead of requiring an instantaneous reversal link \((v_j, v_i) (T=0\) for the bidirectional case) for the link \((v_i, v_j)\), we only need the reversal link \((v_j, v_i)\) to appear within a certain time window or just require \(v_j\) links back to \(v_i\) within a certain time window (The edge \((v_j, v_i)\) may not appear at all).

3.2.4 A Classification of Consensus Results

For better understanding of Assumptions 3.1-3.5, we categorize some of the existing consensus results in Table 3.1. Assumption 3.1 is omitted in the table since it is required in all the results listed.
<table>
<thead>
<tr>
<th>No.</th>
<th>Results</th>
<th>Synchronism</th>
<th>Connectivity</th>
<th>Reversal link</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Th. 2 in [80]</td>
<td>Sync.</td>
<td>A3(a)</td>
<td>NA</td>
</tr>
<tr>
<td>2</td>
<td>Prop. 2 in [120]</td>
<td>Sync.</td>
<td>A3(b)</td>
<td>NA</td>
</tr>
<tr>
<td>3</td>
<td>Prop. 1 in [120]</td>
<td>Sync.</td>
<td>A3(a)</td>
<td>NA</td>
</tr>
<tr>
<td>4</td>
<td>Th. 3.10 in [149]</td>
<td>Sync.</td>
<td>A3(a)</td>
<td>A4(b)</td>
</tr>
<tr>
<td>5</td>
<td>Th. 1 in [121]</td>
<td>Sync.</td>
<td>A3(b)</td>
<td>A4(b)</td>
</tr>
<tr>
<td>6</td>
<td>Th. 2 in [63]</td>
<td>A2(a),(b),(c) with zero time delay</td>
<td>A3(b)</td>
<td>A4(b)</td>
</tr>
<tr>
<td>7</td>
<td>Th. 4 in [29]</td>
<td>A2(a),(c) with zero time delay</td>
<td>A3(b)</td>
<td>A4(b)</td>
</tr>
<tr>
<td>8</td>
<td>Th. 1 in [21]</td>
<td>A2(a),(b),(c) with zero time delay</td>
<td>A3(b)</td>
<td>A4(b)</td>
</tr>
<tr>
<td>9</td>
<td>Th. 3 in [21]</td>
<td>A2(a),(b),(c)</td>
<td>A3(b)</td>
<td>A4(b)</td>
</tr>
<tr>
<td>10</td>
<td>Th. 4 in [21]</td>
<td>A2(a),(b),(c)</td>
<td>A3(b)</td>
<td>A4(b)</td>
</tr>
<tr>
<td>11</td>
<td>Th. 5 in [21]</td>
<td>A2(a),(b),(c)</td>
<td>A3(b)</td>
<td>A4(b)</td>
</tr>
</tbody>
</table>
Several comments are appropriate. First, asynchronous systems with zero time
delay can be mapped to equivalent synchronous systems following the arguments in
[53]. Therefore, Results No. 3 & 4 may be seen as a special case of Result No. 6.
Second, Table 3.1 reveals the fact that the uniform connectivity is necessary under
unidirectional communication but it is not necessary under bidirectional communication. The reason behind this fact is not at all clear from an intuitive perspective.
In Sect. 3.3, we give an explanation to the difference between bidirectional and
unidirectional communication from contractive operators’ point of view. Third, we
utilize the iteration graph in Sect. 3.4.1 to unify various communication assump-
tions. Fourth, Result No. 9 considers the most general case among linear protocols;
notice that all protocols in Table 3.1 are linear protocols. A nonlinear asynchronous
protocol will be introduced in Sect. 3.4.2.

3.3 Bidirectional vs. Unidirectional Communication

In this section, we investigate why different assumptions on connectivity need
to be imposed for bidirectional and unidirectional communication if consensus is
to be achieved. Specifically, we see the consensus problem as a matrix iteration
problem where the notions of paracontraction and pseudocontraction introduced in
Sect. 3.2.1 are useful in proving convergence.

For an easy exposition, we restrict ourselves to synchronous protocols with time-
varying topologies and no time delays. That is, we consider the system updating
equation

\[ x(t + 1) = F(t)x(t) \]

where the matrix \( F(t) \) satisfies Assumption 3.1. It is easily deduced that \( F(t) \) is
nonexpansive with respect to the vector norm \( \| \cdot \|_\infty \). In the bidirectional case, \( F(t) \) is also symmetric. From nonnegative matrix theory [78], we know that the
eigenvalues of $F(t)$ lie in $(-1, +1]$ for all $t$. It is now known that any symmetric matrix $F$ the eigenvalues of which are in $(-1, +1]$ is paracontracting with respect to Euclidian norm [28]. However, $F(t)$ is no longer paracontracting when its symmetry is lost.

Example 2 (partly taken from [175]) (a) For the weight matrix

$$F_1 = \begin{bmatrix} 0.5 & 0.5 & 0 \\ 0.25 & 0.5 & 0.25 \\ 0 & 0.5 & 0.5 \end{bmatrix},$$

induced from the simple communication topology as shown in Fig. 3.1(a), the norm and the set $X^*$ are the same as in Example 1. For any $x$, $P(x) = 0.5(\max_i x_i + \min_i x_i)1$. For $x = [2, 2, 1]^T$, $F_1x = (2, 1.75, 1.5)^T$, $P(x) = 1.5 \cdot 1$, and $P(F_1x) = 1.75 \cdot 1$. Thus the first inequality in (3.4) is strict while the second one is an equality. So this operator is pseudocontractive, not paracontractive.

(b) For an arbitrary $F(t)$, there is no guarantee that it is pseudocontractive, e.g., the weight matrix

$$F_2 = \begin{bmatrix} 1 & 0 & 0 \\ 0.5 & 0.5 & 0 \\ 0 & 0 & 1 \end{bmatrix},$$

which results from the communication topology in Fig. 3.1(b). For $x = [2, 2, 1]^T$, $F_2x = [2, 2, 1]^T$. Thus equalities hold throughout (3.4). $F_2$ is not pseudocontractive but it is nonexpansive.

An alternative proof for Result No. 2 was provided in [201] where the convergence of the paracontracting matrix products was explored. The following theorem in [51] is the key to this development.

**Theorem 3.1** Suppose that a finite set of square matrices of same dimensions \(\{F_1, \cdots, F_r\}\) is paracontracting. Let \(\{I(t)\}_{t=0}^{\infty}\), with \(1 \leq I(t) \leq r\), be a sequence
of integers, and let \( \mathcal{J} \) denote the set of all integers that appear infinitely often in the sequence. Then for all \( x(0) \in \mathbb{R}^n \) the sequence of vectors \( x(t+1) = F_{t(t)}x(t) \) has a limit \( x^* \in \bigcap_{i \in \mathcal{J}} H(F_i) \), where \( H(F) \) denotes the fixed point subspace of a paracontracting matrix \( F \), i.e., its eigenspace associated with the eigenvalue \( 1 \),

\[
H(F) = \{ x | x \in \mathbb{R}^n, Fx = x \}.
\]

Does there exist a result similar to Theorem 3.1 for pseudocontracting matrices? The answer is, fortunately, affirmative.

**Theorem 3.2** ([175]) Let \( \{T_t\} \) be a sequence of nonexpansive operators (with respect to \( \| \cdot \| \) and \( X^* \)), and assume there exists a subsequence \( \{T_{t_i}\} \) which converges to \( T \in T \). If \( T \) is pseudocontractive and uniformly Lipschitz continuous, then for any initial vector \( x(0) \), the sequence of vectors \( x(t+1) = T_{t}x(t), \ t \geq 0 \) converges to some \( x^* \in X^* \).

Note that in our study \( X^* = \{ c1 | c \in \mathbb{R} \} \). We are thus one step away from proving Result No. 3 and it is exactly where Assumption A3(a) (the uniform connectivity) comes into play. Result No. 3 is an immediate result of Theorem 3.2 and the following lemma.

**Lemma 3.1** If Assumption A3(a) is satisfied, then there exists an integer \( B' \) such that the matrix product \( F(t_0 + B')F(t_0 + B' - 1) \cdots F(t_0) \) is pseudocontractive, with respective to \( \| \cdot \|_\infty \) and \( X^* = \{ c1 | c \in \mathbb{R} \} \), for any \( t_0 \geq 0 \).
Lemma 3.2 (Lemma 2 in [80]) Let \( m \geq 2 \) be a positive integer and let \( F_1, F_2, \ldots, F_m \) be nonnegative \( n \times n \) matrices. Suppose that the diagonal elements of all of the \( F_i \) are positive and let \( \alpha \) and \( \beta \) denote the smallest and largest of these, respectively. Then

\[
F_m F_{m-1} \cdots F_1 \geq \left( \frac{\alpha^2}{2\beta} \right)^{(m-1)} (F_m + F_{m-1} + \cdots + F_1).
\]  

(3.11)

Lemma 3.3 (Proposition 3.2 in [175]) Let \( x \) be a vector such that \( \underline{x} < \bar{x} \) with \( \underline{x} = \min_i x_i \) and \( \bar{x} = \max_i x_i \), \( F \) be an irreducible matrix with \( F_{ii} > 0 \) for \( 1 \leq i \leq n \), \( y = Fx \). Then the number of the elements in set \( \{i | y_i = \underline{x} \text{ or } y_i = \bar{x}\} \) is at least one less than the number of the elements in \( \{i | x_i = \underline{x} \text{ or } x_i = \bar{x}\} \).

**Proof** of Lemma 3.1] Assume that there exists an infinite sequence of contiguous, non-empty, bounded time-intervals \([t_{ij}, t_{ij+1})\), \( j \geq 1 \), starting at \( t_{i1} \), with the property that across each such interval, the union of the interaction graphs is strongly connected (Assumption A3(a)).

Let \( H_{t_j} = F(t_{j+1}) \cdots F(t_{j+1})F(t_{ij}) \). By Lemma 3.2 and the strong connectivity of the union of the interaction graphs, it follows that \( H_{t_j} \) is irreducible. It is easy to prove that \( H_{t_j} \) is nonexpansive with positive diagonal elements and \( H_{t_j}1 = 1 \).

Define \( H = H_{t_{n-1}}H_{t_{n-2}} \cdots H_{t_1} \) and \( y = Hx \). Applying Lemma 3.3 repeatedly for \((n-1)\) times, we have at least one set of \( \{i | y_i = \underline{x}\} \) and \( \{i | y_i = \bar{x}\} \) is empty. If, say, \( \{i | y_i = \underline{x}\} \) is empty, then \( y_i > \underline{x} \) for all \( 1 \leq i \leq n \), and furthermore

\[
\|y - P(y)\| = \frac{\max_i y_i - \min_i y_i}{2} \leq \frac{\bar{x} - \min_i y_i}{2} < \frac{\bar{x} - \underline{x}}{2} = \|x - P(x)\|;
\]

therefore \( H \) is pseudocontractive. In other words, \( F(t_0 + B')F(t_0 + B' - 1) \cdots F(t_0) \) is pseudocontractive for \( B' = (n - 1)B \).

Let us briefly summarize what we have presented in this section. In the bidirectional case, the weight matrices are always paracontracting. Theorem 3.1 can be
applied directly to infer the convergence of the consensus processes. In the unidirectional case, the weight matrices are generally not pseudocontracting. In order to use Theorem 3.2, the uniform connectivity condition needs to be imposed so that the matrix products across certain time interval are pseudocontracting.

3.4 Iteration Graph and Nonlinear Asynchronous Consensus Protocols

In the framework of [142], the consensus problem is regarded as a special case of finding common fixed points of a finite set of paracontracting multiple point operators. That is, all the operators are defined on (different) products of $\mathbb{R}^n$. To avoid divergent phenomena, asynchronous iterations which fulfill certain coupling assumptions called confluent are considered. Below we apply this theory of paracontractions and confluence to derive a more general consensus result, which extends Result No. 9 by allowing nonlinear multiple point operators.

For the purpose of self-containedness, we introduce several related definitions below. Let $\mathbb{I}$ be a set of indices, $m \in \mathbb{N}$ a fixed number, and $\mathcal{F} = \{ F^i | i \in \mathbb{I} \}$ be a pool of operators $F^i : D^{m_i} \subset \mathbb{R}^{m_i} \rightarrow D$, where $m_i \in \{ 1, \ldots, m \}$, $\forall i \in \mathbb{I}$, and $D \subset \mathbb{R}^n$ is closed. Furthermore, let $\mathcal{X}_O = \{ x(0), \ldots, x(-M) \} \subset D$ be a given set of vectors. Then, for sequences $\mathcal{I} = I(k)$ ($k = 0, 1, \ldots$) of elements in $\mathbb{I}$, $\mathcal{S} = \{ s^1(k), \ldots, s^{m_i(k)}(k) \}$, $k = 0, 1, \ldots$, of $m_i$-tuple from $\mathbb{N}_0 \cup \{-1, \ldots, -M\}$ with $s^l(k) \leq k$ for all $k \in \mathbb{N}_0$, $l = 1, \ldots, m_i(k)$, we study the asynchronous iteration given by

$$x(k+1) = F^{I(k)}(x(s^1(k)), \ldots, x(s^{m_i(k)}(k))) \text{, } k = 0, 1, \ldots \quad (3.12)$$

An asynchronous iteration corresponding to $\mathcal{F}$, starting with $\mathcal{X}_O$ and defined by $\mathcal{I}$ and $\mathcal{S}$ can be denoted by $(\mathcal{F}, \mathcal{X}_O, \mathcal{I}, \mathcal{S})$. A fixed point $\xi$ of a multiple point operator $F : \mathbb{R}^{nm} \rightarrow \mathbb{R}$ is a vector $\xi \in \mathbb{R}^n$ which satisfies $F(\xi, \ldots, \xi) = \xi$, and a common fixed point of a pool is a fixed point of all its operators in this sense.
3.4.1 Iteration Graph

In essence, the communication assumptions define the coupling among agents or, more generally, the coupling of an iteration process. The existing assumptions often rely on interaction graphs to describe the “spatial” coupling among agents. However, ambiguity arises when asynchronism (e.g., delays) is allowed since the “temporal” coupling cannot be described directly. In the asynchronous setting, it is of importance to differentiate the same agent at different time instants.

To this end, we associate an iteration graph with the asynchronous iteration \((F, X, I, S)\). Every iteration, including initial vectors, gets a vertex, so the set of vertices is \(V = \mathbb{N}_0 \cup \{-1, \ldots, -M\}\). A pair \((k_1, k_2)\) is an element of the set of edges \(E\) in the iteration graph \((V, E)\), if and only if the \(k_1\)th iteration vector is used for the computation of the \(k_2\)th iteration vector.

Below we illustrate the concept of iteration graph via an example. The interaction topologies of a three-agent system at different time instants are shown in Fig. 3.2(a). It is easy to see that if the interaction pattern continues, Assumption 3(b) is satisfied. Let \(y(-1) = x_1(0), y(-2) = x_2(0),\) and \(y(-3) = x_3(0)\). At time \(k = 0\), \(v_2\) communicates with \(v_3\). By construction, we add a vertex 0 and an edge from vertex -2 to vertex 0 in the associated iteration graph, as shown in Fig. 3.2(b). Assume that \(v_3\) and \(v_1\) do not use its own past value. Therefore, we do not add an edge from vertex -3 to vertex 0 in the iteration graph. At time instant \(k = 1\), \(v_2\) uses the value of \(v_1\) and its own past value to update its state, resulting in two edges in the iteration graph.

**Remark 3.1** An analogy to the iteration graph is the reachability graph in the Petri net literature. The reachability graph is used for verification and supervisory control and obtained sometimes via a method called unfolding that simplifies the procedure [13].
Figure 3.2. Interaction topologies of an asynchronous system and its associated iteration graph.

**Definition 3.1 (Confluent asynchronous iteration [142])** Let \((\mathcal{F}, \mathcal{X}_0, \mathcal{I}, \mathcal{S})\) be an synchronous iteration. The iteration graph of \((\mathcal{F}, \mathcal{X}_0, \mathcal{I}, \mathcal{S})\) is the digraph \((V,E)\), whose vertices \(V\) are \(\mathbb{N}_0 \cup \{-1, \ldots, -M\}\), and whose edges \(E\) are given by

\[(k, k_0) \in E, \text{ iff there is an } 1 \leq l \leq m_{I(k_0-1)}, \text{ such that } s^l(k_0 - 1) = k.\]

\((\mathcal{F}, \mathcal{X}_0, \mathcal{I}, \mathcal{S})\) is called confluent, if there are numbers \(n_0 \in \mathbb{N}, \ b \in \mathbb{N}\) and a sequence \(b_k (k = n_0, n_0 + 1, \ldots)\) in \(\mathbb{N}\), such that for all \(k \geq n_0\) the following is true:

(i) For every vertex \(k_0 \geq k\) there is a directed path from \(b_k\) to \(k_0\) in \((V,E)\),

(ii) \(k - b_k \leq B\),

(iii) \(\mathcal{S}\) is regulated (cf. assumption 3.2(a)),

(iv) for every \(i \in \mathbb{I}\) there is a \(c_i \in \mathbb{N}\) so that for all \(k \geq n_0\) there is a vertex \(w^i_k\) in \(V\), which is a successor of \(b_k\) and a predecessor of \(b_{k+c_i}\), and for which is \(I(w^i_k - 1) = i\).

It is worth mentioning that when Assumptions 2(a)(b), 3(b), 4(b) are fulfilled, the associated asynchronous iteration is confluent. Given an arbitrary iteration, there are simple ways to make its implementation confluent [142].

**Remark 3.2** As opposed to the original development in [32], here the whole vector is updated in every iteration step. Also, all components of vectors have the same
delay. This does not impose a restriction since the vectors reduce to a scalar in our study.

3.4.2 Nonlinear Asynchronous Consensus Protocols

Before introducing the main result of the chapter, we need the following definition. Note that paracontracting operator in Definition 3.2(ii) corresponds to pseudocontracting operator as defined in (3.3). (Without much confusion, the original definitions given in [142] are followed for easy reference.)

**Definition 3.2** Let \( F \) be a pool of operations as in Definition 3.1, and \( X = (x^1, \ldots, x^{m_i}) \) an element of \( \mathbb{R}^{n_{m_i}} \).

(i) If for all \( i \in I \), \( X, Y \in D^{m_i} \) and a norm \( \| \cdot \| \)

\[
\|G^i(X) - G^i(Y)\| < \max_j \|x^j - y^j\|
\]

or \( \|G^i(X) - G^i(Y)\| = x^j - y^j, \forall j \in \{1, \ldots, m_i\} \), then \( F \) is called strictly nonexpansive on \( D \).

(ii) If for all \( i \in I \), \( X \in D^{m_i} \) and a norm \( \| \cdot \| \), \( F^i \) is continuous on \( D^{m_i} \), then \( F \) is paracontracting on \( D \), if for any fixed point \( \xi \in \mathbb{R}^n \) of \( F^i \),

\[
\|F^i(X) - \xi\| < \max_j \|x^j - \xi\|
\]

or \( X = (x, \ldots, x) \) and \( x \) is a fixed point of \( F^i \).

It is easy to see that every strictly nonexpansive pool is paracontracting. Moreover, an (e=extended)-paracontracting notion is introduced in [142]. The (e)-paracontracting operators may be discontinuous, and have nonconvex sets of fixed points.

A simplified version of the main result in [142] is now given.

**Theorem 3.3** Let \( F \) be a paracontracting pool on \( D \subset \mathbb{R}^n \), and assume that \( F \) has a common fixed point \( \xi \in D \). Then any confluent asynchronous iterations \( (F, \mathcal{X}_0, \mathcal{I}, \mathcal{S}) \) converges to a common fixed point of \( F \).
With the help of Theorem 3.3, Result No. 9 can be obtained by interpreting the different rows of a stochastic matrix as multiple data operators. To see this, let $f_{im_i(j)}(k)$ be for all $i \in \{1, \ldots, n\}$ the $j$th of $m_i$ nonzero entries in $F(k)$’s $i$th row, or let $m_i(j) = j, \forall j = 1, \ldots, n$, and $m_i = n$, if this row is zero. Then the pool $\mathcal{F} = \{F^i| i = 1, \ldots, Q\}$ ($Q$, the total number of operators, is finite), defined by $F^i(k) : \mathbb{R}^{m_i} \to \mathbb{R}$,

$$F^i(k)(y^1, \ldots, y^{m_i}) := \sum_{j=1}^{m_i} f_{im_i(j)}(k)y^j, \ i = 1, \ldots, Q$$  (3.13)

is strictly nonexpansive on all closed intervals $D \subset \mathbb{R}$, if $F^i(1, \ldots, 1) = 1$.

We are now ready to claim a new consensus result where $F^i$ is allowed to be nonlinear. To avoid confusion, we rewrite $I(k)$ in (3.12) as $p_k$ below.

**Theorem 3.4** Consider the iteration

$$x_i(k+1) = F^i \left(x_1(s^1(k)), x_2(s^2(k)), \ldots, x_n(s^n(k))\right).$$  (3.14)

(i) Assume without loss of generality that the numbering of $s^i(k), \ k = 0, 1, \ldots, $ is chosen in such a manner that all components $x_i(s^i(k))$ in (3.14) themselves are updated at time $s^i(k)$, i.e.,

$$p_{s^i(k)-1} = i, \forall k \in \mathbb{N}, \ i \in \{1, \ldots, n\} \ \text{with} \ s^i(k) \geq 1,$$  (3.15)

and, also w.l.o.g., that all initial vectors are multiples of $1$. Define

$$x(-k) := x_k(0), \ \forall k = 1, \ldots, n,$$

and renumber in this way the elements of the sequences of $s^i(k), \ k = 0, 1, \ldots, l = 1, \ldots, n$, for which $s^i(k) = 0$. Then the asynchronous iteration $(\mathcal{F}, \mathcal{Y}_0, \mathcal{I}, \mathcal{S})$, given by

$$y(k+1) := F^{p_k} \left(y(s^1(k)), \ldots, y(s^{m_{p_k}}(k))\right), \ k = 0, 1, \ldots$$  (3.16)
where $\mathcal{F} = \{F^p_k | k = 0, 1, \ldots \}$ is paracontracting, $\mathcal{I} = p_k$, $k = 0, 1, \ldots$, $\mathcal{S} = \{\tilde{s}^i(k) | k = 0, 1, \ldots; i = 1, \ldots, m_{p_k}\}$ is given by

$$\tilde{s}^i(k) := s^{m_{p_k}(i)}(k), \forall k \in \mathbb{N}_0, i = 1, \ldots, m_{p_k},$$

(3.17)

and $\mathcal{Y}_0$ by $y(-l) := x_1(-l), l = 1, \ldots, n$, generates

$$y(k + 1) = x_{p_k}(k + 1), \forall k \in \mathbb{N}_0.$$  

(3.18)

(ii) The pool $\mathcal{F} = \{F^i : \mathbb{R}^n \to \mathbb{R} | i \in \{1, \ldots, Q\}\}$ is paracontracting and has a common fixed point. Furthermore, there exists an agent $i_0$ which updates its state using a subset of the pool $\mathcal{F}$. Every operator $F^j$ in this subset satisfies the conditions: $F^j(x_{i_0}^1) \neq F^j(x_{i_0}^2)$ for $x_{i_0}^1 \neq x_{i_0}^2$. Assume that $s^{i_0}(k) = \max\{k_0 \leq k | p_{k_0 - 1} = i_0\}$ for all $k > \min\{k_0 \in \mathbb{N}_0 | p_{k_0} = i_0\}$ with $p_k = i_0$.

Then, under Assumptions 2(a)(b), 3(b), 4(b), the nonlinear protocol (3.16) or, equivalently, (3.14) guarantees asymptotic consensus.

Proof: (i) follows by induction on $k$. Using the same argument as in the proof of Theorem 5(v) in [142], it can be shown that the iteration (3.16) is confluent. (ii) is then an immediate result of Theorem 3.3.

Remark 3.3 In Fig. 3.2, the iteration graph is confluent when the agent $v_2$ always uses its own past value for updating. Suppose that no agents use their past values during the process (i.e., dashed edges no longer appear). After removing the dashed edges from Fig. 3.2(b), the iteration graph is no longer confluent since there is no directed path from an odd-numbered vertex to an even-numbered vertex, and vice versa. This shows the necessity of existence of $i_0$ in Theorem 3.4(ii).

Theorem 3.4 is exact, rather than linearized and can be used to study multi-agent systems with nonlinear couplings. Potential applications include distributed time synchronization and rendezvous of multi-robots with nonholonomic constraints.
As an application of Theorem 3.4, we give a formal proof of Theorem 2.2. The idea is to interpret the different rows of a stochastic matrix as multiple data operators; see (3.19).

Proof of Theorem 2.2: Let \( \bar{n} \) be the total number of possible interaction topologies for a graph with \( n \) nodes. The spanning tree assumption implies the interaction graph is confluent. We only need to show that the pool of operator \( \mathcal{F} = \{ F^i | i = 1, \ldots, \bar{n} \} \), defined by

\[
F^i : \mathbb{R}^{m_i} \rightarrow \mathbb{R},
F^i(y^1, \ldots, y^{m_i}) = \sum_{j=1}^{m_i} f_{im_i(j)} y^j, \ i = 1, \ldots, \bar{n}
\] (3.19)

is strictly nonexpansive on all closed intervals \( D \subset \mathbb{R} \), if \( \sum_{j=1}^{m_i} f_{im_i(j)} = 1 \).

To see this, for all \( i \in \{1, \ldots, n\} \), \( x, y \in \mathbb{R}^{m_i} \), we have

\[
|F^i(x) - F^i(y)| \leq \sum_{l=1}^{n} f_l |x^l - y^l| \leq \max_{1 \leq l \leq m_i} |x^l - y^l|
\] (3.20)

and for equality one needs \( x^l - y^l = r = T^i(x - y), \ \forall l = 1, \ldots, m_i \), for some \( r \in \mathbb{R} \).

\( \square \)
CHAPTER 4

DECENTRALIZED FORMATION TRACKING OF MULTIPLE-VEHICLE SYSTEMS

4.1 Introduction

Control problem involving mobile vehicles/robots have attracted considerable attention in the control community during the past decade. One of the basic motion tasks assigned to a mobile vehicle may be formulated as following a given trajectory [108, 46]. The trajectory tracking problem was globally solved in [157] by using a time-varying continuous feedback law, and in [107, 37, 134] through the use of dynamic feedback linearization. The backstepping technique for trajectory tracking of nonholonomic systems in chained form was developed in [82, 91]. In the special case when the vehicle model has a cascaded structure, the higher dimensional problem can be decomposed into several lower dimensional problems that are easier to solve [136].

An extension to the traditional trajectory tracking problem is that of coordinated tracking or formation tracking (see Fig. 4.1). The problem is often formulated as to find a coordinated control scheme for multiple robots that make them maintain some given, possibly time-varying, formation while executing a given task as a group. The possible tasks could range from exploration of unknown environments where an increase in numbers could potentially reduce the exploration time, navigation in hostile environments where multiple robots make the system redundant and thus
Figure 4.1. Six vehicles perform a formation tracking task.

robust, to coordinated path following; see recent survey papers [160, 33].

In formation control of multi-vehicle systems, different control topologies can be adopted depending on applications. There may exist one or more leaders in the group with other vehicles following one or more leaders in a specified way. In many scenarios, vehicles have limited communication ability. Since global information is often not available to each vehicle, distributed controllers using only the local information are desirable. One approach to distributed formation control is to represent formations using the vectors of relative positions of neighboring vehicles and the use of consensus-based controllers with input bias [55, 102].

In this chapter, we study the formation tracking problem for a group of vehicles/robots using the consensus-based controllers combined with the cascade approach [136]. The idea is to specify a reference path for a given, nonphysical point. Then a multiple vehicle formation, defined with respect to the real vehicles as well as to the nonphysical virtual leader, should be maintained at the same time as the virtual leader tracks its reference trajectory. The vehicles exchange information according to a communication digraph, $G$. Similar to the tracking controller in [136], the controller for each vehicle can be decomposed to two “sub-controllers,”
one for positioning and one for orientation. Different from the traditional single vehicle tracking case, each vehicle uses information from its neighbors in the communication digraph to determine the reference velocities and stay at their designation in the formation. Based on nonlinear synchronization results [197], we prove that consensus-based formation tracking can be achieved as long as the formation graph had a spanning tree and the controller parameters are large enough (Their lower bounds relates to a quantity determined by Laplacian matrices of formation graphs).

Related work includes [90, 147, 59, 64, 168]. In [90], the vehicle dynamics were assumed to be linear and formation control design was based on algebraic graph theory. In [147], output feedback linearization control was combined with a second-order (linear) consensus controller to coordinate the movement of multiple mobile robots. The problem of vehicles moving in a formation along constant or periodic trajectories was formulated as a nonlinear output regulation (servomechanism) problem in [59]. The solutions adopted in [168, 64] for coordinated path following control of multiple marine vessels or wheeled robots built on Lyapunov techniques, where path following and inter-vehicle coordination were decoupled.

The contributions of this work are: 1) The consensus-based formation tracking controller for nonlinear vehicles is novel and its stability properties are examined using cascaded systems theory and nonlinear synchronization theory; 2) Global results allow us to consider a large class of trajectories with arbitrary (rigid) formation patterns and initial conditions.
4.2 Preliminaries

4.2.1 Tracking Control of Mobile Vehicles

A kinematic model of a wheeled mobile robot with two degrees of freedom is given by the following equations

\[
\begin{align*}
\dot{x} &= v \cos \theta, \\
\dot{y} &= v \sin \theta, \\
\dot{\theta} &= \omega, \\
\end{align*}
\]  
(4.1)

where the forward velocity \( v \) and the angular velocity \( \omega \) are considered as inputs, \((x, y)\) is the center of the rear axis of the vehicle, and \( \theta \) is the angle between heading direction and \( x \)-axis (see Fig. 4.2).

For time varying reference trajectory tracking, the reference trajectory must be selected to satisfy the nonholonomic constraint. The reference trajectory is hence generated using a virtual reference robot [85] which moves according to the model

\[
\begin{align*}
\dot{x}_r &= v_r \cos \theta_r, \\
\dot{y}_r &= v_r \sin \theta_r, \\
\dot{\theta}_r &= \omega_r, \\
\end{align*}
\]  
(4.2)

where \([x_r y_r \theta_r]^T\) is the reference posture obtained from the virtual vehicle. Following

Figure 4.2. Mobile robots and the error dynamics.
we define the error coordinates (cf. Fig. 4.2)

\[ p_e = \begin{bmatrix} x_e \\ y_e \\ \theta_e \end{bmatrix} = \begin{bmatrix} \cos \theta & \sin \theta & 0 \\ -\sin \theta & \cos \theta & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x_r - x \\ y_r - y \\ \theta_r - \theta \end{bmatrix}. \quad (4.3) \]

It can be verified that in these coordinates the error dynamics become

\[ \dot{p}_e = \begin{bmatrix} \dot{x}_e \\ \dot{y}_e \\ \dot{\theta}_e \end{bmatrix} = \begin{bmatrix} \omega y_e - v + v_r \cos \theta_e \\ -\omega x_e + v_r \sin \theta_e \\ \omega_r - \omega \end{bmatrix}. \quad (4.4) \]

The aim of (single robot) trajectory tracking is to find appropriate velocity control laws \( v \) and \( \omega \) of the form

\[
\begin{align*}
  v &= v(t, x_e, y_e, \theta_e) \\
  \omega &= \omega(t, x_e, y_e, \theta_e)
\end{align*}
\quad (4.5)
\]

such that the closed-loop trajectories of (4.4) & (4.5) are stable in some sense (e.g., uniform globally asymptotically stable).

As discussed in Sect. 4.1, there are numerous solutions to this problem in the continuous time domain. Here, we revisit the cascaded approach proposed in [136]. Let us first introduce the notion of globally \( K \)-exponential stability.

**Definition 4.1** A continuous function \( \alpha : [0, a) \to [0, \infty) \) is said to belong to class \( K \) if it is strictly increasing and \( \alpha(0) = 0 \).

**Definition 4.2** A continuous function \( \beta : [0, a) \times [0, \infty) \to [0, \infty) \) is said to belong to class \( KL \) if for each fixed \( s \) the mapping \( \beta(r, s) \) belongs to class \( K \) with respect to \( r \), and for each fixed \( r \) the mapping \( \beta(r, s) \) is decreasing with respect to \( s \) and \( \beta(r, s) \to 0 \) as \( s \to \infty \).
Definition 4.3  Consider the system
\[ \dot{x} = g(t, x), \quad g(t, 0) = 0 \quad \forall t \geq 0 \]  \hspace{1cm} (4.6)
where \( g(t, x) \) is piecewise continuous in \( t \) and locally Lipschitz in \( x \).

We call the system (4.6) globally \( K \)-exponentially stable if there exist \( \xi > 0 \) and a class \( K \) function \( k(\cdot) \) such that
\[ \|x(t)\| \leq k(\|x(t)\|)e^{-\xi(t-t_0)}. \]

Theorem 4.1 ([136])  Consider the system (4.4) in closed-loop with the controller
\[
\begin{align*}
v &= v_r + c_2 x_e, \\
\omega &= \omega_r + c_1 \theta_e,
\end{align*}
\]  \hspace{1cm} (4.7)
where \( c_1 > 0 \) \( c_2 > 0 \). If \( \omega_r(t), \dot{\omega}_r(t), \text{ and } v_r(t) \) are bounded and there exist \( \delta \) and \( k \) such that
\[
\int_t^{t+\delta} \omega_r(\tau)^2 d\tau \geq k, \quad \forall t \geq t_0
\]  \hspace{1cm} (4.8)
then the closed-loop system (4.4) \& (4.7), written compactly as
\[
\dot{p}_e = h(x_e, y_e, \theta_e)|_{v_r, \omega_r} = h(p_e)|_{v_r, \omega_r}
\]  \hspace{1cm} (4.9)
is globally \( K \)-exponentially stable. \( \Box \)

In the above, the subscriptions for \( h(\cdot)|_{v_r, \omega_r} \) mean that the error dynamics are defined relative to reference velocities \( v_r \) and \( \omega_r \). The tracking condition (4.8) implies that the reference trajectories should not converge to a point (or straight line). This also relates to the well-known persistence-of-excitation condition in adaptive control theory.

Note that control laws in (4.7) are linear with respect to \( x_e \) and \( \theta_e \). This is critical in designing consensus-based controller for multiple vehicle formation tracking as we shall see below.
4.2.2 Formation Graphs

We consider formations that can be represented by acyclic directed graphs. In these graphs, the agents involved are identified by vertices, and the leader-following relationships by (directed) edges. The orientation of each edge distinguishes the leader from the follower. Follower controllers implement static state feedback-control laws that depend on the state of the particular follower and the states of its leaders.

**Definition 4.4 ([180])** A formation control graph $G = (V, E, D)$ is a directed acyclic graph consisting of the following.

- A finite set $V = \{v_1, \ldots, v_N\}$ of $N$ vertices and a map assigning to each vertex a control system $\dot{x}_i = f_i(t, x_i, u_i)$ where $x_i \in \mathbb{R}^n$ and $u_i \in \mathbb{R}^m$.
- An edge set encoding leader-follower relationships between agents. The ordered pair $(v_i, v_j) \triangleq e_{ij}$ belongs to $E$ if $u_j$ depends on the state of agent $i$, $x_i$.
- A collection $D = \{d_{ij}\}$ of edge specifications, defining control objectives (set-points) for each $j$: $(v_i, v_j) \in E$ for some $v_i \in V$.

For agent $j$, the tails of all incoming edges to vertex represent leaders of $j$, and their set is denoted by $L_j \subset V$. Formation leaders (vertices of in-degree zero) regulate their behavior so that the formation may achieve some group objectives, such as navigation in obstacle environments or tracking reference paths.

Given a specification $d_{kj}$ on edge $(v_k, v_j) \in E$, a setpoint for agent $j$ can be expressed as $x^r_j = x_k - d_{kj}$. For agents with multiple leaders, the specification redundancy can be resolved by projecting the incoming edges specifications into orthogonal components

$$x^r_j = \sum_{k \in L_j} S_{kj}(x_k - d_{kj}) \quad (4.10)$$

where $S_{kj}$ are projection matrices with $\sum_k \text{rank}(S_{kj}) = n$. Then the error for the closed-loop system of vehicle $j$ is defined to be the deviation from the prescribed
setpoint $\tilde{x}_j \triangleq x^r_j - x_j$, and the formation error vector is constructed by stacking the errors of all followers

$$\tilde{x} \triangleq [\cdots \tilde{x} \cdots]^T, \ v_j \in V \setminus L_F.$$ 

4.2.3 Input-to-state Stability on Formation Graphs

Formation input-to-stability (ISS) is a stability notion for interconnected systems that can be used in the performance analysis and design of formations [179]. It yields quantitative measures of the performance of leader-follower formation structures in terms of stability.

**Definition 4.5 (Formation ISS)** A formation is called input-to-state stable if there is a class $KL$ function $\beta$ and a class $K$ function $\gamma$ such that for any any initial formation error $\tilde{x}(0)$ and for any bounded inputs of the formation leaders $u_i(t)$, the formation error satisfies:

$$\|\tilde{x}(t)\| \leq \beta(\|\tilde{x}(0)\|, t) + \sum_{t \in L} \gamma(t)(\sup_{[0,t]} \|u_t\|)$$ (4.11)

The ISS property of three primitive subgraphs was presented in [179]. In this chapter, we will use these results to derive a bound for the formation tracking error that depends on the leaders input and the initial errors.

4.2.4 Formation Control as Consensus Problems

In networks of agents (or dynamic systems), “consensus” means to reach an agreement regarding a certain quantity of interest that depends on the state of all agents. A “consensus algorithm” (or protocol) is an interaction rule that specifies the information exchange between an agent and all of its neighbors on the network. For a recent literature review on consensus problems, see [54, 151, 130].
A simple consensus algorithm to reach an agreement regarding the state of \( n \) integrator agents with dynamics \( \dot{x}_i = u_i \) can be expressed as an \( n \)th-order linear system on a graph [130]:

\[
\dot{x}_i = \sum_{j \in L_i} (x_j(t) - x_i(t)) + b_i(t), \quad x_i(0) \in \mathbb{R}, \quad b_i(t) = 0.
\] (4.12)

Formation tracking requires consent and collaboration of every vehicle in the formation. Consider the simple vehicle dynamics \( \dot{x}_i = u_i \) (\( x_i \) is the position of vehicle \( i \)) and the following distributed protocol

\[
\dot{x}_i = \sum_{j \in L_i} (x_j - d_{ji} - x_i) = \sum_{j \in L_i} (x_j - x_i) + b_i
\] (4.13)

with input bias \( b_i = -\sum_{j \in L_i} d_{ji} \). It can be shown [130] that protocol (4.13) minimizes the local “formation error” term

\[
C_i(x) = \sum_{j \in L_i} \|x_j - d_{ji} - x_i\|.
\] (4.14)

Since non-zero bias does not affect the stability property of (4.12), relative-position based formation control is a consensus problem. (For a more precise exposition, see Theorems 4.3 4.4.)

For more complicated nonlinear vehicle dynamics (4.1), the simple linear protocol (4.12) is not sufficient. In the following, we will introduce results on synchronization in networks of nonlinear dynamical systems. As we shall see later, the formation tracking problems can be studied in this framework.

4.2.5 Synchronization in networks of nonlinear dynamical systems

**Definition 4.6** Given a matrix \( V \in \mathbb{R}^{n \times n} \), a function \( f(y, t) : \mathbb{R}^{n+1} \to \mathbb{R}^n \) is \( V \)-uniformly decreasing if \( (y - z)^T V (f(y, t) - f(z, t)) \leq -\mu \|y - z\|^2 \) for some \( \mu > 0 \) and all \( y, z \in \mathbb{R}^n \) and \( t \in \mathbb{R} \).
Note that a differentiable function $f(y, t)$ is $V$-uniformly decreasing if and only if $V(\partial f(y)/\partial y) + \delta I$ for some $\delta > 0$ and all $y, t$. Consider the following synchronization result for the coupled network of identical dynamical systems with state equations:

$$
\dot{x} = (f(x_1, t), \ldots, f(x_N, t))^T + (C(t) \otimes D(t))x + u(t),
$$

where $x = (x_1, \ldots, x_N)^T$, $u = (u_1, \ldots, u_N)^T$ and $C(t)$ is a zero sums matrix for each $t$. $C \otimes D$ is the Kronecker product of matrices $C$ and $D$.

**Theorem 4.2** ([197]) Let $Y(t)$ be an $n$ by $n$ time-varying matrix and $V$ be an $n$ by $n$ symmetric positive definite matrix such that $f(x, t) + Y(t)x$ is $V$-uniformly decreasing. Then the network of coupled dynamical systems in (4.15) synchronizes in the sense that $\|x_i - x_j\| \to 0$ as $t \to \infty$ for all $i, j$ if the following two conditions are satisfied:

(i) $\lim_{t \to \infty} \|u_i - u_j\| = 0$ for all $i, j$.

(ii) There exists an $N$ by $N$ symmetric irreducible zero row sums matrix $U$ with nonpositive off-diagonal elements such that

$$(U \otimes V)(C(t) \otimes D(t) - I \otimes Y(t)) \leq 0$$

(4.16)

for all $t$.

4.3 Basic Formation Tracking Controller

The control objective is to solve a formation tacking problem for $N$ vehicles. This implies that each vehicle must converge to and stay at their designation in the formation while the formation as a whole follows a virtual vehicle.

Equipped with the results presented in the previous section, we first construct a basic formation tracking controller (FTC) from (4.7). Let $d_{ri}=[d_{x_{ri}} \ d_{y_{ri}}]^T$ denote the formation specification on edge $(v_r, v_i)$. By virtue of linear structures of (4.7), we propose:
Figure 4.3. Formation tracking using baseline FTC. The reference vehicle sends to vehicle \( \textit{i} \) the formation specification \( d_{ri} \) as well as the reference velocities \( v_r \) and \( \omega_r \).

**Basic FTC for vehicle \( \textit{i} \):**

\[
\begin{align*}
    v_i &= v_r + c_2 x_{ei}, \\
    \omega_i &= \omega_r + c_1 \theta_{ei},
\end{align*}
\]

(4.17)

where \( c_1 > 0, c_2 > 0 \) and

\[
\begin{align*}
    p_{ei} &= [x_{ei} y_{ei} \theta_{ei}]^T \\
    &= \begin{bmatrix}
        \cos \theta_i & \sin \theta_i & 0 \\
        -\sin \theta_i & \cos \theta_i & 0 \\
        0 & 0 & 1
    \end{bmatrix}
    \begin{bmatrix}
        x_r - x_i - d_{x_{ri}} \\
        y_r - y_i - d_{y_{ri}} \\
        \theta_r - \theta_i
    \end{bmatrix}.
\end{align*}
\]

(4.18)

**Remark 4.1** It is not required to have constraints for every pair of vehicles. We need only a sufficient number of constraints which uniquely determine the formation.

**Theorem 4.3** The basic FTC (4.17) and (4.18) solves the formation tracking problem.

**Proof:** By Theorem 4.1, every vehicle \( \textit{i} \) follows the virtual (or leader) vehicle (thus the desired trajectory) with a formation constraint \( d_{ri} \) on edge \((v_r, v_i)\). Thus all vehicles tracks the reference trajectory while staying in formation, which is specified by formation constraints \( d_{ri} \)'s as shown in Fig. 4.3.

\[ \square \]
Corollary 4.1 Suppose only vehicle 1 follows the virtual vehicle. The composite system with inputs $v_r$ and $\omega_r$ and states $\tilde{x}_1 = [x_{e1} \ y_{e1} \ \theta_{e1}]^T$ is globally $K$-exponentially stable and therefore formation input-to-state stable (see Appendix).

Example 4.1 [Basic FTC] Assume that we have a system consisting of three vehicles, which are required to move in some predefined formation pattern. First, as in [59], we will consider the case of moving in a triangle formation along a circle. That is, the virtual (or reference) vehicle dynamics are given by: $x_r = v_r \cos(\omega_r t) + x_{r0}$, $y_r = v_r \sin(\omega_r t) + y_{r0}$, where $v_r$ is the reference forward velocity, $\omega_r$ the reference angular velocity, and $[x_{r0} \ y_{r0}]^T$ the initial offsets.

Assume that the parameters $v_r = 10$, $\omega_r = 0.2$, $[x_{r0} \ y_{r0}]^T = [-25 \ 0]^T$. In our simulations we used an isosceles right triangle with sides equal to $3\sqrt{2}$, $3\sqrt{2}$, and 6. Also, fix the position of the virtual leader at the vertex with the right angle. Then, from the above constraints the required (fixed) formation specifications for the vehicles are given by

$$d_{r1} = \begin{bmatrix} 0 \\ 0 \end{bmatrix}, \quad d_{r2} = \begin{bmatrix} 3 \\ 3 \end{bmatrix}, \quad d_{r3} = \begin{bmatrix} 3 \\ -3 \end{bmatrix}. $$

For FTC we chose the parameters as $c_1 = 0.3$ and $c_2 = 0.5$. Fig. 4.4 shows the trajectories of the system for about 100 seconds. Initially the vehicles are not in the required formation; however, they form the formation quite fast ($K$-exponentially fast) while following the reference trajectory (solid line in the figure). Fig. 4.4 shows the control signals $v$ and $\omega$ for each vehicle.

\[\square\]

4.4 Consensus-based Formation Tracking Controller

The basic FTC has the advantage that it is simple and leads to globally stabilizing controllers. A disadvantage, however, is that it requires every vehicle to get
Figure 4.4. Circular motion of three vehicles with a triangle formation. Initial vehicle postures are: $[-8 - 9 \ 3\pi/5]^T$ for vehicle 1 (denoted as *); $[-15 - 20 \ \pi/2]^T$ for vehicle 2 (□); $[-10 - 15 \ \pi/3]^T$ for vehicle 3 (◦).
Figure 4.5. Control signals $v$ and $\omega$ for Virtual vehicle: solid line; Vehicle 1: dotted line; Vehicle 2: dashed line; and Vehicle 3: dot-dash line.

access to the reference velocities $v_r$ and $\omega_r$. This further implies that the reference vehicle needs to establish direct communication links with all other vehicles in the group, which may not be practical in some applications.

In a more general setting, we assume that only a subset of vehicles (leaders) have direct access to the reference velocities. Other vehicles (followers) use their neighboring leaders’ information to accomplish the formation tracking task. In this case, formation tracking controllers operate in a decentralized fashion since only neighboring leaders’ information has been used.
Consensus-based FTC for vehicle $i$

\[
\begin{align*}
v_i &= v_{ri} + c_2 x_e_i + \sum_{j \in L_i} a_{ij} (x_e_i - x_e_j), \\
\omega_i &= \omega_{ri} + c_1 \theta_e_i + \sum_{j \in L_i} a_{ij} (\theta_e_i - \theta_e_j), \\
\dot{v}_{ri} &= \sum_{j \in L_i} a_{ij} (v_r_j - v_r_i), \\
\dot{\omega}_{ri} &= \sum_{j \in L_i} a_{ij} (\omega_r_j - \omega_r_i)
\end{align*}
\]

(4.19)

where

\[
 p_{e_i} = \begin{bmatrix} x_{e_i} \\
 y_{e_i} \\
 \theta_{e_i} \end{bmatrix} = \begin{bmatrix} \cos \theta_i & \sin \theta_i & 0 \\
 -\sin \theta_i & \cos \theta_i & 0 \\
 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x_i^r - x_i \\
 y_i^r - y_i \\
 \theta_i^r - \theta_i \end{bmatrix}.
\]

and $a_{ij}$ represents relative confidence of agent $i$ in the information state of agent $j$.

**Remark 4.2** As one can see from (4.19), the communication between vehicles is local and distributed, in the sense that each vehicle receives the posture and velocity information only from its neighboring leaders.

We have the following theorem regarding the stability of the consensus-based FTC.

**Theorem 4.4** The consensus-based FTC (4.19) solves the formation tracking problem if the formation graph $G$ has a spanning tree and the controller parameters $c_1, c_2 > 0$ are large enough. Lower bounds for $c_1$ and $c_2$ are related to the Laplacian matrix for $G$.

**Proof:** Let $L_G$ be the Laplacian matrix induced by the formation graph $G$ and it is defined by

\[
(L_G)_{ij} = \begin{cases}
\sum_{k=1, k \neq i}^N a_{ik}, & j = i \\
-a_{ij}, & j \neq i
\end{cases}
\]
We will write $P_e = [p_{e_1}, \ldots, p_{e_N}]^T \in \mathbb{R}^{3N}$, $[V_r \ \Omega_r]^T = [v_{r_1}, \ldots, v_{r_N}, \omega_{r_1}, \ldots, \omega_{r_N}]^T \in \mathbb{R}^{2N}$. The closed loop system (4.19)-(4.4) for all vehicles can be expressed in a compact form as

\[ \dot{P}_e = \begin{bmatrix} h(p_{e_1})|_{v_{r_1}, \omega_{r_1}} \\ \vdots \\ h(p_{e_N})|_{v_{r_N}, \omega_{r_N}} \end{bmatrix} + (-L_G \otimes D)P_e, \quad (4.20) \]

\[ \begin{bmatrix} \dot{V}_r \\ \dot{\Omega}_r \end{bmatrix} = (-L_G \otimes I_2) \begin{bmatrix} V_r \\ \Omega_r \end{bmatrix}. \quad (4.21) \]

where

\[ D = \begin{bmatrix} -1 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & -1 \end{bmatrix} \quad (4.22) \]

describes the specific coupling between two vehicles.

It can be seen that (4.21) is in the form of linear consensus algorithms. Since the formation graph is acyclic and has a rooted spanning tree (with the root corresponding to the virtual vehicle), the reference velocities (coordination variables) $v_{r_i}(t)$ and $\omega_{r_i}(t)$ for any vehicle $i$ in the group will approach to $v_r(t)$ and $\omega_r(t)$, respectively [151, 130]. (It is important for the formation graph to be acyclic such that each vehicle can follow Lipschitz-continuous root reference velocities with arbitrarily small errors. For general graphs with loops, the consensus algorithms have band-limited properties [132].)

We thus re-write (4.20) as

\[ \dot{P}_e = \begin{bmatrix} h(p_{e_1})|_{v_r, \omega_r} \\ \vdots \\ h(p_{e_N})|_{v_r, \omega_r} \end{bmatrix} + (-L_G \otimes D)P_e + \begin{bmatrix} \phi_1(t) \\ \vdots \\ \phi_N(t) \end{bmatrix} \quad (4.23) \]
and $\phi_i(t) \to 0$ as $t \to \infty$. The functions $\phi_i$ can be considered as residual errors that occurred when replacing $v_{ri}$ and $\omega_{ri}$ in (4.20) with $v_r$ and $\omega_r$, respectively.

Now (4.23) is in the same form of (4.16). We further set $Y = \alpha D$ so that $h(p_e) + \alpha D p_e$ is $V$-uniformly decreasing (see Lemma 11 in [199]) provided that $c_1 - \alpha > 0$ and $c_2 - \alpha > 0$. Theorem 4.2 says that (4.23) synchronizes if there exists a symmetric zero row sums matrix $U$ with nonpositive off-diagonal elements such that $(U \otimes V)(-L_G \otimes D - I \otimes Y) \leq 0$. Since $VD \leq 0$ and $Y = \alpha D$, this is equivalent to

$$U(-L_G - \alpha I) \geq 0.$$  \hspace{1cm} (4.24)

Let $\mu(-L_G)$ be the supremum of all real numbers $\alpha$ such that $U(-L_G - \alpha I) \geq 0$. It was shown in [198] that $\mu(-L_G)$ exists for constant row sums matrices and can be computed by a sequence of semidefinite programming problems. Choose $c_1$ and $c_2$ to be large enough such that

$$\min\{c_1, c_2\} > \mu(-L_G)$$  \hspace{1cm} (4.25)

and $\|P_{ei} - P_{ej}\| \to 0$ as $t \to \infty$ for all $i, j$. The spanning tree assumption guarantees that at least one vehicle, say vehicle 1, has its error coordinates $P_{e1} \to 0$ as $t \to \infty$ by Theorem 4.3. This further implies that $P_{ei} \to 0$, as $t \to \infty$ for all $i$. The proof is complete.

In particular, an upper bound for $\mu(-L_G)$ is given by $\mu_2(-L_G) = \min \text{Re}(\lambda)$ where $\text{Re}(\lambda)$ is the real part of $\lambda$, the eigenvalues of $-L_G$ that do not correspond to the eigenvector $e$. It suffices to make $\min\{c_1, c_2\} > \mu_2(-L_G)$.

\textbf{Example 4.2} In this example, we chose virtual vehicle dynamics of a sinusoidal form: $(x_r(t), y_r(t)) = (t, \sin(t))$. The acyclic formation graph with formation specifications is shown in Fig. 4.6.
Figure 4.6. A formation graph with formation specifications on edges

The (unweighted) Laplacian matrix corresponds to Fig. 4.6 is given by:

\[ L_G = \begin{bmatrix}
1 & -1 & 0 & 0 \\
0 & 2 & -1 & -1 \\
0 & 0 & 1 & -1 \\
0 & 0 & 0 & 0
\end{bmatrix}. \]  
(4.26)

Since \( \mu_2(-L_G) = -2 \) we used consensus-based FTC (4.19) with positive \( c_1, c_2 \), say \( c_1 = 0.3 \) and \( c_2 = 0.5 \). As shown in Fig. 4.7, successful formation tracking with a desired triangle formation is achieved. Vehicle control signals \( v_i \)'s and \( \omega_i \)'s are shown in Fig. 4.8.

4.5 Formation ISS

To study performance for the closed-loop system (4.4) and (4.19), we restrict ourselves to the class of formation structures that can be constructed by composition and superposition of primitive types of subgraphs. We consider three primitive subgraphs of diameter two [179]: the cascade interconnection of three vehicles (Fig.
Figure 4.7. Tracking a sinusoidal trajectory in a triangle formation. Initial vehicle postures are: $[12 \ 12 \ 0]^T$ for vehicle 1 (denoted as $*$); $[-15 \ -20 \ \pi/4]^T$ for vehicle 2 (□); $[-10 \ 15 \ -\pi/4]^T$ for vehicle 3 (○).
Figure 4.8. Vehicle control signals $v_i$'s and $\omega_i$'s.

4.9(a)), the parallel interconnection of four vehicles (Fig. 4.9(b)) and the multiple-leader interconnection (Fig. 4.9(c)).

Consider five vehicles $i$, $j$, $k$, $m$, and $n$. First let vehicles $i$, $j$ and $k$ be connected in cascade (Fig. 4.9(a)), where $i$ is assigned to follow $j$, $j$ should follow $k$ and $k$ may have to follow some reference trajectory $y(t)$. Define the errors for agents $i$, $j$, and $k$ as:

$$
\tilde{x}_i \triangleq x_i^r - x_i \equiv x_j - d_{ji} - x_i
$$

$$
\tilde{x}_j \triangleq x_j^r - x_j \equiv x_k - d_{kj} - x_j
$$

$$
\tilde{x}_k \triangleq x_k^r - x_k \equiv x_m - d_{mk} - x_k
$$

Suppose that control laws $u_i = u_i(x_i, x_j)$, $u_j = u_j(x_j, x_k)$ and $u_k = u_k(x_k, x_m)$ are designed so that each follower can track its leader, and the closed loop error
dynamics can take form:

\[
\begin{align*}
\dot{x}_i &= \tilde{f}_i(t, \tilde{x}_i, \tilde{x}_j) \\
\dot{x}_j &= \tilde{f}_j(t, \tilde{x}_j, \tilde{x}_k) \\
\dot{x}_k &= f_k(t, \tilde{x}_k)
\end{align*}
\]  

(4.27) (4.28) (4.29)

Proposition 4.1 (Nonlinear Cascade) If (4.27) and (4.28) are ISS with respect to \( \tilde{x}_j \) and \( \tilde{x}_k \) respectively, then the \( \tilde{x}_g = (\tilde{x}_i, \tilde{x}_j) \)-system is ISS with respect to \( \tilde{x}_k \).

Proof: The proposition follows from Corollary 4.1 and application of Theorem 4.5 in the Appendix.

Consider the parallel interconnection of Fig. 4.9(b). Agents \( i \) and \( j \) are assigned to follow agent \( k \), and \( k \) follows the group leader \( m \). Application of the feedback laws \( u_i = u_i(x_i, x_k) \), \( u_j = u_j(x_j, x_k) \), \( u_k = u_k(x_k, x_m) \) can bring the closed loop error dynamics to the form:

\[
\begin{align*}
\dot{x}_i &= \tilde{f}_i(t, \tilde{x}_i, \tilde{x}_k) \\
\dot{x}_j &= \tilde{f}_j(t, \tilde{x}_j, \tilde{x}_k) \\
\dot{x}_k &= \tilde{f}_k(t, \tilde{x}_k, \tilde{x}_m)
\end{align*}
\]  

(4.30) (4.31) (4.32)

81
where $\tilde{x}_i = x_k - d_{ki} - x_i$, $\tilde{x}_j = x_k - d_{kj} - x_j$ and $\tilde{x}_k = x_m - d_{mk} - x_k$.

**Proposition 4.2 (Nonlinear Parallel)** Let (4.30) with input $\tilde{x}_k$ and (4.31) with input $\tilde{x}_k$ be ISS and (4.32) be ISS with respect to $\tilde{x}_m$. Then the composed $\tilde{x}_g = (\tilde{x}_i, \tilde{x}_j, \tilde{x}_k)$ -system is ISS with respect to $\tilde{x}_m$.

The multiple leader interconnection of Fig. 4.9(c) is realized by the feedback control laws $u_i = u_i(x_i, x_j, x_k)$, $u_j = u_j(x_j, x_n)$, $u_k = u_k(x_k, x_m)$ that can bring the closed loop error dynamics to the following form:

\[
\begin{align*}
\dot{\tilde{x}}_i &= \tilde{f}_i(t, \tilde{x}_i, \tilde{x}_j, \tilde{x}_k) \\
\dot{\tilde{x}}_j &= \tilde{f}_j(t, \tilde{x}_j, \tilde{x}_n) \\
\dot{\tilde{x}}_k &= \tilde{f}_k(t, \tilde{x}_k, \tilde{x}_m)
\end{align*}
\] (4.33) (4.34) (4.35)

**Proposition 4.3 (Nonlinear Multiple-Leader)** Let (4.33) be ISS with respect to $\tilde{x}_j$ and $\tilde{x}_k$, (4.34) be ISS with respect to $\tilde{x}_n$ and (4.35) be ISS with respect to $\tilde{x}_m$. Then the composed $\tilde{x}_g = (\tilde{x}_i, \tilde{x}_j, \tilde{x}_k)$ -system is ISS with respect to $\tilde{x}_n$ and $\tilde{x}_m$.

**Remark 4.3** Since all the assumptions in Propositions 3.2 and 3.3 in [179] are satisfied for mobile vehicles under study, identical statements Propositions 4.2 and 4.3 are given in the above without proofs.

4.6 Conclusions

This chapter addressed the formation tracking problem for multiple mobile vehicles with nonholonomic constraints. We developed a basic formation tracking controller (FTC) as well as a consensus-based one using only neighboring leaders information. The stability properties of the multiple vehicle system in closed loop with these FTCs were studied using cascaded systems theory and nonlinear synchronization theory. In particular, we established connections between stability of
consensus-based FTC and Laplacian matrices for formation graphs. Our simple formation tracking strategy holds great potential to be extended to the case of air and marine vehicles.

Appendix: Input-to-State Stability and Cascade-Connected Systems [113]

Consider the nonlinear system

$$\dot{x} = f(x, u) \quad (4.36)$$

where $f : D \times D_u \to \mathbb{R}^n$ is locally Lipschitz in $x$ and $u$. The sets $D$ and $D_u$ are defined by $D = \{x \in \mathbb{R}^n : \|x\| < r\}$, $D_u = \{u \in \mathbb{R}^m : \sup_{t>0} \|u(t)\| = \|u\|_{L_\infty} < r_u\}$.

**Definition 4.7** The system (4.36) is said to be locally input-to-state stable (ISS) if there exist a KL function $\beta$, a class $K$ function $\gamma$ and constants $k_1, k_2 \in \mathbb{R}^+$ such that

$$\|x(t)\| \leq \beta(\|x_0\|, t) + \gamma(\|u_T(\cdot)\|_{L_\infty}), \; \forall t \geq 0, \; 0 \leq T \leq t \quad (4.37)$$

for all $x_0 \in D$ and $u \in D_u$ satisfying: $\|x_0\| < k_1$ and $\sup_{t>0} \|u_T(t)\| = \|u_T\|_{L_\infty} < k_2$, $0 \leq T \leq t$. It is said to be input-to-state stable, or globally ISS if $D = \mathbb{R}^n$, $D_u = \mathbb{R}^m$ and (4.37) is satisfied for any initial state and any bounded input $u$.

Consider the composite systems shown in Fig. 4.10, where $\Sigma_1$ and $\Sigma_2$ are given by

$$\Sigma_1 : \quad \dot{x} = f(x, z) \quad (4.38)$$

$$\Sigma_2 : \quad \dot{z} = g(z, u) \quad (4.39)$$
where $\Sigma_2$ is the system with input $u$ and state $z$. The state of $\Sigma_2$ serves as input to the system $\Sigma_1$.

**Theorem 4.5** Consider the cascade interconnection of the systems $\Sigma_1$ and $\Sigma_2$. If both systems are input-to-state stable, then the composite system $\Sigma : u \rightarrow [x \ z]^T$ is input-to-state stable.
CHAPTER 5

STABILIZATION AND PERFORMANCE ANALYSIS FOR A CLASS OF SWITCHED SYSTEMS

5.1 Introduction

In recent years, considerable research efforts have been devoted to the study of switched systems. The motivation for studying switched systems comes from the fact that many practical systems are inherently multimodal [41], and the fact that some of intelligent control methods are based on the idea of switching between different controllers [126, 129, 47, 76]. The existence of systems that cannot be asymptotically stabilized by a single static continuous feedback controller [27] also motivates the study. A survey of basic problems in stability and design of switched systems is given in [97]. For a brief review on recent interests of switched systems in the consensus problem, see Section 1.2.

One of the basic problems is to find conditions which guarantee that switched systems are asymptotically stable under arbitrary switching sequences. This property, although conservative, is desirable in some cases. For example, we may design separate controller to meet different criteria, while all these closed-loop systems (plant plus different controllers) satisfy the condition for stability under arbitrary switching rule. Therefore, in the implementation of the multi-controller structure, one may switch among these controllers only by the performance requirements, without worrying about the stability of the whole systems. This gives us more freedom
when synthesizing switching sequence for the performance concerns, and better fault tolerance property as well. As an example, an observer, which is stable for arbitrary switching sequences, is designed in [5] for linear systems with randomly-switching measurement equations.

Stability analysis of switched systems is usually carried out in the Lyapunov framework [26, 47]. For switched linear systems, stability under arbitrary switching is equivalent to the existence of a common Lyapunov function, which is not necessarily quadratic [97, 41]. Although progress has been recently made [128, 165, 34, 98], finding a common Lyapunov function is still an open problem. Quite often, a linear matrix inequality (LMI) problem formulation is used to obtain sufficient stability conditions by constructing a set of quadratic Lyapunov-like functions [84, 138]. More recently, the switched Lyapunov function (SLF) method and less conservative LMI based conditions were developed in [36] for stability analysis and control design for switched linear systems. The approach covers common quadratic Lyapunov functions as a special case. For switched linear systems, the existence of SLF is a weaker condition than the solvability of Lie algebra, which implies the existence of a common quadratic Lyapunov function [1]. The Lie algebra approach, however, can be generalized to study switched nonlinear systems [112] while the extension of SLF approach to the nonlinear context is not straightforward. In the SLF method, we also assume that the SLF strictly decrease along the solutions of the systems for all time instances. This restrictive assumption can be relaxed for certain classes of switched linear systems. One can deduce asymptotic stability using multiple Lyapunov functions whose Lie derivative are only negative semi-definite [74]. The SLF method has been applied to solve different problems of switched linear systems [35, 202, 81]. In particular, the input-output performance problem was studied in [35]. There are some related works in the literature on analyzing the input-output properties.
of switched systems. For example, the $\mathcal{L}_2$ analysis [206] and the $l^\infty$ disturbance attenuation problem [99] have been studied.

Other approaches in the stability study of switched systems, not necessarily under arbitrary switching, include (average) dwell-time approach [75], min-projection approach [137], conic switching law [203], etc.

The outline of the chapter is as follows. In Section 5.2, we explore the stability equivalence between linear time-varying (LTV) systems and switched linear (SL) systems. A new proof is provided based on Caratheodary's Theorem in convex analysis. By using the existing results on robust stability, necessary and sufficient conditions for asymptotic stability of switched systems under arbitrary switching have been obtained in different ways. Connections to the literature were also provided when deemed necessary. In Section 5.3, a new LMI based stability test for the existence of switched Lyapunov functions is first developed. If a switched Lyapunov function exists, asymptotic stability of the switched system also implies its exponential stability. An LMI optimization problem is then formulated to find a bound on the decay rate of the system. To attain the bound, state feedback control gains are designed. Using the same framework and the well-known $S$-procedure, a generalized sufficient LMI condition is obtained which guarantees a $\gamma$-performance of the closed-loop switched systems subject to input disturbances.

Notations

$\mathbb{Z}$ is the set of integer numbers. Let $\mathbb{R}^{m\times n}$ denote the set of $m \times n$ real matrices and $\mathbb{R}^+ = [0, +\infty)$. $\mathbb{S}^n$ denotes the set of $n \times n$ real symmetric matrices and $\mathbb{S}_+^n$, the set of $n \times n$ real symmetric positive-definite matrices. $\lambda(M)$ stands for the eigenvalue of a matrix $M$. $M^T$ is the transpose of the matrix $M$. $M > 0$ ($M < 0$) means that $M$ is positive definite (negative definite). The matrix norm $\|A\|$, defined on $\mathbb{R}^{m\times n}$, and induced by the vector norm $\|x\|$ in $\mathbb{R}^n$, is defined as
\[ \| A \| = \max \| x \| = \max_{1 \leq i \leq n} |x_i|, \] 
In particular, for \( \| x \|_\infty = \max_{1 \leq i \leq n} |x_i|, \) we have \( \| A \|_\infty = \max_{1 \leq i \leq n} \left\{ \sum_{j=1}^{n} |a_{ij}| \right\}. \) For a given set \( A = \{A_1, \cdots, A_q\} \) of constant matrices \( A_i \in \mathbb{R}^{n \times n}, i = 1, \cdots, q, \) we denote the convex hull of this set (convex matrix polyhedron in \( \mathbb{R}^{n \times n} \)) by \( \text{co} A = \text{co}(A_1, \cdots, A_q). \)

5.2 A Necessary and Sufficient Condition

In this section, necessary and sufficient conditions for asymptotic stability of discrete-time SL systems under arbitrary switching sequences are derived. It is first shown that, for the discrete-time case, the asymptotic stability of SL systems under arbitrary switchings is equivalent to the robust stability of polytopic uncertain linear time-varying (LTV) systems. This fact allows extending all of the techniques developed for LTV systems (e.g., see [118, 119]) to SL systems, therefore rendering the LTV framework a useful companion for investigating properties of SL systems. As an immediate application, “new” necessary and sufficient stability conditions for SL systems under arbitrary switching sequences are readily obtained from robust stability theory. In particular, the existence of a polyhedral Lyapunov function, which strictly decreases along the solutions of the system, is necessary and sufficient for the asymptotic stability of a SL system. We further apply the equivalence result to study several special cases in SL systems and new simple proofs are provided for some well-known stability results. This section also presents, in a unified way, a number of results scattered in the literature.

5.2.1 Stability Notions of Switched Linear Systems and Linear Time-Varying Systems

Consider the (autonomous) SL system

\[ x(t+1) = A_\sigma x(t), \quad x \in \mathbb{R}^n. \] (5.1)
where $t \in \mathbb{Z}^+ = \{0, 1, \cdots \}$ and the switching signal $\sigma : \mathbb{Z}^+ \rightarrow Q = \{1, \cdots, q\}$ takes value from the index set $Q$. A switching signal is called arbitrary, if there is no restriction on it, i.e., all possible switching sequences can be followed. The finite set $Q = \{1, \cdots, q\}$ is called the set of modes of the SL system.

Related to (5.1), we define the discrete LTV system with polytopic uncertainties as

$$x(t+1) = A(t)x(t), \quad A(t) \in \text{co} \mathcal{A}. \quad (5.2)$$

**Definition 5.1** (Asymptotic stability of SL systems [96]) The system (5.1) is said to be asymptotically stable if there exists a positive constant $\delta_2$ and a class $\mathcal{KL}$ function $\beta_2$ such that for all switching signals $\sigma$ the solutions of (5.1) with $\|x(0)\| \leq \delta_2$ satisfy $\|x(t)\| \leq \beta_2(\|x(0)\|, t)$, $\forall t \in \mathbb{Z}^+$.

**Definition 5.2** (Robust stability of LTV systems [9]) The system (5.2) is said to be robustly stable with respect to the set $\mathcal{A}$ if the zero solution $x(t) \equiv 0$ of this system is asymptotically stable, i.e., if there exists a positive constant $\delta_1$ and a class $\mathcal{KL}$ function $\beta_1$ such that for any time-varying matrix $A(t) \in \mathcal{A}$ the solutions of (5.2) with $\|x(0)\| \leq \delta_1$ satisfy $\|x(t)\| \leq \beta_1(\|x(0)\|, t)$, $\forall t \in \mathbb{Z}^+$.

Definitions 5.1 & 5.2 are formulated with the fact that for SL system (5.1) under arbitrary switching and LTV system (5.2) the concepts of local and global asymptotic stability are equivalent [16]. Moreover, asymptotic stability in this case is equivalent to exponential stability is also well known. It is obvious that the robust stability of (5.2) implies the asymptotic stability of (5.1).

### 5.2.2 Equivalence between Stability Concepts

In this section, we show that the asymptotic stability of SL system (5.1) is equivalent to the robust stability of LTV system (5.2). We do not claim that this result
is new since it has already been verified in the literature, e.g., [16, 71]. However, it seems to be little referencing to the robust stability literature from the switched systems area. The emphasis here is to make this connection concrete based on a new proof with the aid of the convex analysis. The existing robust stability results are therefore used to obtain “new” necessary and sufficient conditions for SL systems.

Consider the following two time-invariant discrete inclusions DIs:

\[ x(k+1) \in G(x(k)), \quad G(x) = \{ y : y = Ax, A \in \mathcal{A} \}, \quad (5.3) \]

\[ x(k+1) \in \co G(x(k)), \]

\[ \co G(x) = \{ y : y = Ax, A \in \co \mathcal{A} \}, \quad (5.4) \]

where \( \mathcal{A} \subset \mathbb{R}^{n \times n} \) is a compact set in \( \mathbb{R}^{n \times n} \). Let \( X(k, k_0, x_0) \) and \( Y(k, k_0, x_0) \) denote the reachability sets from an initial point \( x(k_0) = x_0 \) at time \( k \geq k_0 \) for difference inclusions (5.3) and (5.4), respectively. We have the following results.

**Lemma 5.1** For any \( x_0 \) and \( k = k_0, k_0 + 1, \cdots \), we have

\[ X(k, k_0, x_0) \subseteq Y(k, k_0, x_0) \subseteq \co X(k, k_0, x_0). \quad (5.5) \]

See Appendix.

**Lemma 5.2** The zero solution \( x(k) = 0 \) of difference inclusion (5.3) is asymptotically stable if and only if the zero solution of (5.4) is asymptotically stable.

The sufficiency follows from the left inclusion of (5.5). Necessity follows from the right inclusion of (5.5).

The rationale of the present section hinges on the following theorem providing the stability equivalence between LTV systems and SL systems under arbitrary switching:
Theorem 5.1 The system (5.1) is asymptotically stable if and only if the system (5.2) is robustly stable.

It is easy to know that SL system (5.1) and LTV system (5.2) are equivalent to the DIs (5.3) and (5.4), respectively [16, 71]. The equivalence is regarded in the sense of coincidence of the sets of solutions of (5.1) and of (5.3), and the sets of solutions of (5.2) and of (5.4).

Let the compact set $\mathcal{A}$ in (5.3) and (5.4) taken to be $\mathcal{A} = \{A_1, A_2, \cdots, A_q\}$. The theorem follows from Lemma 5.2.

Theorem 5.1 says that the asymptotic stability of (5.1) also implies the robust stability of (5.2), i.e., for robust stability of a LTV system, it suffices to only consider its vertex dynamics as an arbitrarily switching system. Although based on different arguments, this equivalence has been implied in the literature, e.g., [8, 16, 71]. What is new here is that a novel proof is given in a constructive way.

Theorem 5.1 provides a simple connecting link between the systems (5.1) and (5.2). The use of this theorem enables us to obtain “new” stability criteria for SL systems using the existing robust stability results [9, 117].

Theorem 5.2 The system (5.1) is asymptotically stable if and only if

i) There exists a finite integer $k \geq 1$, such that $\|A_{i_1}A_{i_2} \cdots A_{i_k}\| < 1$, for any $A_{i_j} \in \mathcal{A}, \ j = 1, \ldots, k$;

ii) There exists a finite integer $m \geq n$, a full column rank matrix $H \in \mathbb{R}^{m \times n}$, and $m \times m$ matrices $U_i, i = 1, \cdots, q$, satisfying the conditions $\|U_i\|_{\infty} < 1, i = 1, \cdots, q$, such that the matrix relations $HA_i = U_iH, i = 1, \cdots, q$ are satisfied [117].

iii) There exists a full column rank matrix $H \in \mathbb{R}^{m \times n}$ (for some integer $m \geq n$) and a constant $\theta$ (0 < $\theta$ < 1) such that a piecewise linear Lyapunov function of the
polyhedral vector norm type

\[ V_H(x) = \|x\|_H = \|Hx\|_\infty \quad (5.6) \]

satisfies the inequality

\[ \max_{y \in G(x)} V_H(y) \leq \theta V_H(x), \ x \in \mathbb{R}^n \quad (5.7) \]

where the set \( G(x) \subset \mathbb{R}^n \) is defined at each point \( x \in \mathbb{R}^n \) by \( G(x) = \{ y : y = Ax, A \in \mathcal{A} \} \) [117].

In general, the conditions in Theorem 5.2 i) and ii) can be checked by numerical methods [9, 140, 141]. Although they do not give a conclusive test for asymptotic stability of (5.1) since nothing is said about how large an integer \( k \) or \( m \) should be to satisfy the conditions of the theorem, it is known that the conditions must be satisfied with finite \( k \) or \( m \) whenever the system (5.1) is asymptotically stable.

**Remark 5.1** It may be said that the problem of testing affirmatively for asymptotic stability of (5.1) is not in the class of polynomial-time testable problems [20]. It is yet unclear whether the problem is decidable. The related problem of determining if all trajectories of a SL system are bounded is known to be undecidable [19]. It is also shown recently that SL systems that are periodically stable may be unstable [17, 18].

**Remark 5.2** The asymptotic stability of (5.1) with respect to the set \( \mathcal{A} \) is related to the notion of the *joint spectral radius* of a set of matrices, which was first defined in [152] and has appeared in a number of different contexts; see, e.g., [172] or [40] for recent surveys. One particular occurrence in the present context is discussed in [19, 20, 18].
From Theorem 5.2 i) with \( k = 1 \) and the fact that

\[
\rho(A) = \max_{1 \leq j \leq n} |\lambda_j(A)| \leq \|A\|,
\]

the following corollaries can be easily obtained.

**Corollary 5.1** The system (5.1) is asymptotically stable if there exists in \( \mathbb{R}^n \) a vector norm \( \| \cdot \| \) such that the induced matrix norm \( \|A_i\| < 1 \), \( \forall A_i \in \mathcal{A} \).

**Corollary 5.2** For the asymptotic stability of (5.1), it is necessary that every matrix \( A_i \in \mathcal{A} \) is Schur stable, i.e., \( \rho(A_i) < 1 \).

### 5.2.3 Application of the Equivalence Result for Special Cases

With Theorem 5.2 and Corollaries 5.1 & 5.2, we are ready to show that some well known stability results for SL systems can also be proved by using robust stability theory for LTV systems; see also [119].

**Necessary and Sufficient Conditions**

The next corollary has been stated in a number of papers, e.g., [128, 58]. Here we give a new proof by using Theorem 5.2.

**Corollary 5.3** If \( \mathcal{A} \) is a finite set of commuting matrices, then the system (5.1) is asymptotic stable if and only if every \( A_i, A_i \in \mathcal{A} \), is Schur stable.

**Proof:** Necessity follows from Corollary 5.2. For sufficiency, note that in case of pairwise commutative \( A_i, i = 1, \cdots, q \), any matrix product \( \pi_k \overset{\Delta}{=} A_{i_1}A_{i_2}\cdots A_{i_k} \), for any \( A_{i_j} \in \mathcal{A} \), can be written as

\[
\pi_k = A_1^{k_1}A_2^{k_2}\cdots A_q^{k_q}, \quad \sum_{j=1}^{q} k_j = k.
\]
Since \( \max_{1 \leq j \leq q} \rho(A_j) < 1 \) for any induced matrix norm there exist constant \( L \geq 1 \) and \( \xi \) such that

\[
\max_{1 \leq j \leq q} \| A_{kj}^k \| \leq L \xi^k, \quad k = 0, 1, 2, \ldots .
\]

From (5.9), we have

\[
\| \pi_k \| \leq L^q \xi^k, \quad \text{for all } \pi_k, \quad k = 0, 1, 2, \ldots .
\]

Choose the integer \( k > q \ln L/|\ln \xi| \), we arrive at the condition in Theorem 5.2 i), thus completing the proof.

\[\square\]

**Remark 5.3** It can be shown that if \( A_i \) are pairwise commutative, then there exists a nonsingular matrix \( U \) such that \( U^{-1}A_iU \) are upper (lower) triangular [125]. Furthermore, if \( A_i \) are Schur stable, a common quadratic Lyapunov function (CQLF) \( V(x) = x^TPx \) \( (P > 0) \) exists for the system (5.1). Although a number of numerical solutions have long been obtained [25], a general analytic solution to the existence or nonexistence of CQLFs has yet to be obtained [3].

It is well known that the spectral norm equals to the spectral radius for a normal matrix, i.e., \( \| A \|_2 = \rho(A) \). (A square matrix \( A \) is a normal matrix if \( AA^T = A^T A \).)

If the spectral norm is specifically chosen as the matrix norm in Corollary 5.1, the next corollary then follows from Corollaries 5.1 & 5.2.

**Corollary 5.4** *If \( A \) is a finite set of normal matrices, then the system (5.1) is asymptotic stable if and only if every \( A_i, A_i \in A, \) is Schur stable.*

**Remark 5.4** Note that symmetric matrices are normal. The results in [205] thus become special cases of the above Corollary; see also [192]. In this case, the conditions assure the existence of the common Lyapunov function \( V(x) = \| x \|_2^2 = x^T x \)
for the system (5.1).
Some Sufficient Conditions

In this section, we introduce several sufficient conditions for asymptotic stability of (5.1). The proposed conditions are special corollaries of the results established in Section 5.2.2, because these conditions are reduced to checking the Schur stability of some particular test matrix.

Let $|A|$ denote the matrix obtained from $A$ by taking the absolute value of all entries. For two $m \times n$ matrices $A$ and $B$, $A \leq B$ denotes an element-wise inequality.

Let us associate with a given set of matrices $\{A_j, j = 1, \cdots, q\}$, a nonnegative majorant matrix

$$
\hat{A} = \max_{A_j \in A} \{|A_j|\},
$$

(5.10)

where the maximum is understood to be element-wise.

**Theorem 5.3** The system (5.1) is asymptotically stable, if the matrix $\hat{A}$ in (5.10) is Schur stable, i.e., $\rho(\hat{A}) < 1$. The condition is also necessary for each of the following cases

**Case 1)** There exists at least one matrix $A_i \in A$ such that $A_i = \hat{A}$ or $A_i = -\hat{A}$; cf. [124].

**Case 2)** There exists at least one matrix $A_i \in A$ such that $|A_i| = \hat{A}$ and $\hat{A}$ is a Morishima matrix. A matrix is a Morishima matrix if and only if $SAS = |A|$ for some matrix $S = \text{diag} \{s_1, \ldots, s_n\}$ with $s_i = \pm 1, i = 1, \cdots, n$; cf. [163, 15].

**Case 3)** The matrices $A_i \in A$ are either all upper triangular or all lower triangular; cf. [67].

**Proof:** The proof is similar to those of Theorems 5.1 and 5.2 in [119], thus is omitted here. □
Note that the statement of Case 3) of Theorem 5.3 is a discrete-time counterpart of Prop. 2.8 in [98]. It is also worth pointing out that it is a nontrivial matter to find a basis in which all matrices take the triangular form (*simultaneous triangularization*) or even decide whether such a basis exists. Some conditions under which the $A$-matrices can be simultaneously triangularized using a non-singular transformation are given in [165]. For an extensive discussion on this subject, we refer the reader to [144]. Moreover, pairwise triangularizability is not sufficient to guarantee the asymptotic stability of switched systems [164].

5.3 Stability and Performance Analysis: A Switched Lyapunov Function Approach

Motivated by the work in [36], we combine the SLF method with Finsler’s Lemma [167, 44] to study stability and control design problems with performance analysis for discrete-time arbitrarily switching linear systems. Two performance measures considered in this section are the decay rate and the input-output performance. First, a new LMI-based necessary and sufficient condition, which generalizes the one in [36], is obtained to check the existence of a SLF. Our new method is conceptually simple. Here, difference equations are considered as constraints and these dynamical constraints are incorporated into the stability analysis condition through the use of matrix Lagrange multipliers. The key idea is to increase the dimension of the LMIs and to introduce new matrix variables, allowing extra degree of freedom for stability analysis and control design. We show that if a SLF exists, the switched system is not only asymptotically stable but also exponentially stable. Of particular interest is the formulation of an LMI optimization problem to find a sharp estimate of the decay rate to the origin for switched linear systems. To attain a bound on the decay rate, switched state feedback control design is investigated. By switched control design, we mean the design of state feedback control gains for each subsystem such that
the closed-loop switched system is asymptotically (or exponentially) stable. The closed-loop switched system is the one corresponding to the closed-loop subsystems under an arbitrary switching rule. Finally, we study the robustness of the feedback control law when the system is subject to input disturbances from a γ-performance point of view, which provides an upper bound on the worst case energy amplitude gain for switched systems over all possible inputs and switching signals. Using the same mathematical tool and the well-known S-procedure [24], a new LMI-based sufficient condition is obtained to ensure the asymptotic stability of the switched system while satisfying a γ-performance condition. Better performance level than the one in [35] is guaranteed due to extra matrix variables introduced in the new LMI condition. Most of the introductory material on SLF can be found in [36].

5.3.1 Problem Formulation

Consider the class of switched linear systems with input

\[ x(t + 1) = A_{\sigma(t)} x(t) + B_{\sigma(t)} u(t), \quad t \in \mathbb{Z}_+ \]  \hspace{1cm} (5.11)

where \( x(t) \in \mathbb{R}^n \) is the state, \( u(t) \in \mathbb{R}^m \) is the control input, and \( \mathbb{Z}_+ = \{0, 1, 2, \ldots\} \). The switching signal \( \sigma(t) : \mathbb{Z}_+ \rightarrow \mathcal{I} = \{1, \ldots, N\} \) and is unknown \textit{a priori}. (For notational simplicity, we may not explicitly mention the time-dependence of the switching signal below.) Here, \((A_i, B_i) (i \in \mathcal{I})\) are constant matrices of appropriate dimensions denoting the subsystems, and \( N \geq 2 \) is the number of subsystems.

We are interested in the following important questions.

1) \textit{Sufficient conditions for asymptotic stability:} Is it possible to specify conditions such that the autonomous switched system (when \( u = 0 \)) is asymptotically stable?

2) \textit{Decay rate:} If the answer to Q1) is yes, can a bound on the decay rate to the origin be found?
Q3) *Control design:* If the answer to Q2) is yes, does a (state feedback) control $u(t)$ exist such that the bound is actually attained for the closed loop system?

Q4) *Robustness of the control law:* Is the control robust when subject to input disturbances? In the sequel we show that answers to all the above questions are indeed yes.

### 5.3.2 Stability Analysis

In this section, we investigate the stability of the origin of an autonomous switched system. The asymptotic stability of the system is verified by means of a set of LMIs formulated in terms of subsystem $A$-matrices. If feasible, these LMIs provide a set of Lyapunov matrices that can be combined to form a switched quadratic Lyapunov function.

The autonomous switched system is given by

$$x(t + 1) = A_{\sigma(t)}x(t).$$

(5.12)

Define the indication function

$$\xi(t) = [\xi_1(t), \ldots, \xi_N(t)]^T$$

(5.13)

with

$$\xi_i(t) = \begin{cases} 1, & \sigma(t) = i \\ 0, & \text{otherwise} \end{cases}$$

Then, the switched system (5.12) can also be written as

$$x(t + 1) = \sum_{i=1}^{N} \xi_i(t)A_i x(t).$$

(5.14)

To check asymptotic stability of system (5.12), a SLF with a structure similar to that of the system description was used [36]:

$$V(t, x(t)) = x^T(t) \left( \sum_{i=1}^{N} \xi_i(t)P_i \right) x(t),$$

(5.15)
where $P_i \in S^n_+$, $(i = 1, \ldots, N)$. If such a positive-definite Lyapunov function exists and

$$\Delta V(t, x(t)) = V(t + 1, x(t + 1)) - V(t, x(t))$$

(5.16)

is negative definite along the solutions of (5.12), then the origin of the switched system (5.12) is asymptotically stable.

In the following, a new LMI-based necessary and sufficient condition is obtained by combining SLF method with Finsler’s Lemma [167, 44]. This condition is more general than those conditions of Theorem 2 in [36] and no matrix inversion is involved in the construction of SLF.

We first introduce Finsler’s Lemma, which has been previously used in the control literature mainly with the purpose of eliminating design variables in matrix inequalities. In this context, Finsler’s Lemma is usually referred to as Elimination Lemma.

**Lemma 5.3 (Finsler’s Lemma)** Let $x \in \mathbb{R}^n$, $P \in S^n$, and $H \in \mathbb{R}^{m \times n}$ such that $\text{rank}(H) = r < n$. The following statements are equivalent:

1) $x^T P x < 0$, $\forall H x = 0$, $x \neq 0$.

2) $\exists X \in \mathbb{R}^{n \times m}$: $P + XH + H^TX^T < 0$.

In Lemma 5.3, item 1) has a *constrained* quadratic form in $\mathbb{R}^n$ while item 2) provides an *unconstrained* quadratic form, where the constraint is taken into account by introducing multiplier $X$.

Recall that the requirement $\Delta V(t, x(t)) < 0$, $\forall x(t) \neq 0$ can be stated as
\[ \exists P_i, P_j \in S^+_n \text{ such that} \]
\[
\begin{bmatrix}
  x(t)^T & x(t+1)^T
\end{bmatrix}
\begin{bmatrix}
  -P_i & 0 \\
  0 & P_j
\end{bmatrix}
\begin{bmatrix}
  x(t) \\
  x(t+1)
\end{bmatrix} < 0,
\]
\[ \forall \begin{bmatrix}
  A_i & -I
\end{bmatrix}
\begin{bmatrix}
  x(t) \\
  x(t+1)
\end{bmatrix} = 0, \quad \begin{bmatrix}
  x(t) \\
  x(t+1)
\end{bmatrix} \neq 0,
\]
assuming that \( \sigma(t) = i \) and \( \sigma(t+1) = j \).

It now becomes clear that Lemma 5.3 can be applied to the switched system under study. In the new procedure, the dynamic difference equations that characterize the system are seen as constraints, which are naturally incorporated into the stability condition using Finsler’s Lemma. In contrast with standard space methods, where stability is carried in the space of the state vector, the stability test is generated in the enlarged space containing both the state and its time difference.

**Theorem 5.4** There exists a Lyapunov function of the form (5.15) whose difference is negative definite, proving asymptotic stability of (5.12) if and only if there exist \( P_i \in S^+_n \), and matrices \( F_i, G_i \in \mathbb{R}^{n \times n} \) \((i = 1, \ldots, N)\), satisfying
\[
\begin{bmatrix}
  A_i F_i^T + F_i A_i^T - P_i & A_i G_i - F_i \\
  G_i^T A_i^T - F_i^T & P_j - G_i - G_i^T
\end{bmatrix} < 0,
\]
\[ \forall (i, j) \in \mathcal{I} \times \mathcal{I}. \quad (5.17) \]

The Lyapunov function is then given by (5.15).

**Proof:** Apply Lemma 5.3 with
\[
x \leftarrow \begin{bmatrix}
  x(t) \\
  x(t+1)
\end{bmatrix}, \quad P \leftarrow \begin{bmatrix}
  -P_i & 0 \\
  0 & P_j
\end{bmatrix},
\]
\[
H^T \leftarrow \begin{bmatrix}
  A_i^T \\
  -I
\end{bmatrix}, \quad X \leftarrow \begin{bmatrix}
  F_i \\
  G_i^T
\end{bmatrix},
\]
\[ P_i \in S^+_n, \quad F_i, G_i \in \mathbb{R}^{n \times n} \quad (i \in \mathcal{I}). \]
The equivalence between asymptotic stability of (5.12) and the following feasibility test is then established:

$$\exists P_i \in S_+^n, F_i, G_i \in \mathbb{R}^{n \times n} \ (i \in \mathcal{I}):$$

$$
\begin{bmatrix}
A_i^T F_i^T + F_i A_i - P_i & A_i^T G_i - F_i \\
G_i^T A_i - F_i^T & P_j - G_i - G_i^T
\end{bmatrix} < 0.
$$

(5.18)

The result follows by transposing $A_i$ in (5.18). \hfill \Box

The key idea behind Theorem 5.4 is to increase the dimension of the LMIs and to introduce new matrix variables $F_i$ and $G_i$, here identified as Lagrange multipliers, allowing some degree of freedom to verify (5.15) and (5.16). With special choices of $F_i$ and $G_i$, we have the following corollary.

**Corollary 5.5** The following statements are equivalent.

i) There exists a Lyapunov function of the form (5.15) whose difference is negative definite, proving asymptotic stability of (5.12).

ii) There exist $P_i \in S_+^n, \ (i = 1, \ldots, N)$, satisfying

$$
\begin{bmatrix}
P_i & A_i^T P_j \\
P_j A_i & P_j
\end{bmatrix} > 0, \ \forall (i, j) \in \mathcal{I} \times \mathcal{I}.
$$

(5.19)

The Lyapunov function is then given by (5.15).

iii) There exist $P_i \in S_+^n$ and $G_i \in \mathbb{R}^{n \times n} \ (i = 1, \ldots, N)$, satisfying

$$
\begin{bmatrix}
-P_i & A_i G_i \\
G_i^T A_i^T & P_j - G_i - G_i^T
\end{bmatrix} < 0, \ \forall (i, j) \in \mathcal{I} \times \mathcal{I}.
$$

(5.20)

The Lyapunov function is given by (5.15).

**Proof:** The equivalence of i)-iii) follows from Theorem 5.4 by making $F_i = G_i = 0$ for item ii) and $F_i = 0$ for item iii), respectively in condition (5.18). \hfill \Box

It is not difficult to see that Corollary 5.5 is essentially equivalent to Theorem 2 in [36] while the proof here is more straightforward. Moreover, with no matrix
TABLE 5.1

NUMERICAL COMPLEXITY ASSOCIATED WITH THREE STABILITY TESTS

<table>
<thead>
<tr>
<th>Stability Tests</th>
<th>$K$ (scalar variables)</th>
<th>$L$ (LMI rows)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cor. 5.5-ii)</td>
<td>$\frac{Nn(n+1)}{2}$</td>
<td>$Nn + 2N^2n$</td>
</tr>
<tr>
<td>Cor. 5.5-iii)</td>
<td>$\frac{Nn(n+1)}{2} + Nn^2$</td>
<td>$Nn + 2N^2n$</td>
</tr>
<tr>
<td>Theorem 5.4</td>
<td>$\frac{Nn(n+1)}{2} + 2Nn^2$</td>
<td>$Nn + 2N^2n$</td>
</tr>
</tbody>
</table>

inversion involved in the Lyapunov function, Theorem 5.4 allows us to formulate an LMI optimization problem to find a sharp estimate of exponential convergence rate of (5.12) as illustrated in Section 5.3.3, provided that (5.17) is feasible.

Remark 5.5 The numerical complexity associated with the LMI conditions can be computed in terms of the number $K$ of scalar variables and number $L$ of LMI rows. As discussed in [24], the number of floating point operation or the time required to test the feasibility of the set of LMIs is proportional to $K^3L$. Table 5.1 shows $K$ and $L$ as a function of $n$ (states) and $N$ (subsystems) for three tests presented here. (For a practical purpose, only those instances with $N \leq 10$, $n \leq 10$ is tractable.) In the case of restrictive switching signals, we can modify these conditions or invoke the $\mathcal{S}$-procedure to improve the conservatism of these conditions. Take for instance, a system which does not allow arbitrary transitions between subsystems will have the set of all ordered pairs $(i, j)$ of subsystem indices much smaller than $I \times I$. 

102
5.3.3 Attainable Bounds on Decay Rate

A Bound on Decay Rate

From Theorem 5.4, the asymptotic stability of a switched linear system under arbitrary switching can be checked with the feasibility test (5.17). Indeed, we can say more: if the switched system (5.12) is asymptotically stable, it is also exponentially stable about the origin, i.e., \( \exists \kappa > 0 \) and \( 0 \leq \xi < 1 \), such that

\[
\| x(t) \| \leq \kappa \cdot \xi^t \| x(0) \|. \tag{5.21}
\]

for all initial conditions \( x(0) \) and for all \( t \geq 0 \). In fact, the Lyapunov function (5.15) is positive definite, decrescent, and radially unbounded since \( V(t, x(t)) = 0, \forall t \geq 0 \), and

\[
\eta \| x(t) \|^2 \leq V(t, x(t)) \leq \rho \| x(t) \|^2 \tag{5.22}
\]

for all \( x(t) \in \mathbb{R}^n \) with \( \eta = \min_{i \in \mathcal{I}} \lambda_{\text{min}}(P_i) \) and \( \rho = \max_{i \in \mathcal{I}} \lambda_{\text{max}}(P_i) \) positive scalars. Furthermore,

\[
\Delta V \leq -\nu \| x(t) \|^2 \tag{5.23}
\]

with \( \nu = \min_{(i,j) \in \mathcal{I} \times \mathcal{I}} \lambda_{\text{min}}(P_i - A_i^T P_j A_i) < \rho \). With the above observations, the exponential stability of (5.12) about the origin immediately follows from the well-known fact [155]:

**Theorem 5.5** The sequence \( x(t) \) is exponentially stable about the origin if there exists a Lyapunov function \( V(t, x(t)) \) such that \( \forall t \geq 0 \)

\[
\eta \| x(t) \|^2 \leq V(t, x(t)) \leq \rho \| x(t) \|^2,
\]

\[
V(t+1, x(t+1)) - V(t, x(t)) \leq -\nu \| x(t) \|^2, \tag{5.24}
\]

\( \eta, \rho, \nu > 0 \). Then \( \| x(t) \| \leq \kappa \cdot \xi^t \| x(0) \| \), where \( \kappa^2 = \rho/\eta \) and \( \xi^2 = 1 - \nu/\rho \).
We obtain the following corollary.

**Corollary 5.6** If the LMIs given in Theorem 5.4 or Corollary 5.5 are satisfied, the autonomous switched system (5.12) is exponentially stable about the origin.

In the case of verifying exponential stability of (5.12), it may be desirable not only find feasible solutions to (5.20) or (5.17) but to search for solutions that give an estimate of the decay rate $\xi$ in Theorem 5.5. To this end, we want to maximize the ratio $\nu/\rho$. Since $\Delta V = x^T(t) (A_i^T P_j A_i - P_i) x(t)$, constraints (5.22) and (5.23) can be rewritten as:

$$\eta I < P_i < \rho I,$$

$$
\begin{bmatrix}
A_i F_i^T + F_i A_i^T - P_i + \nu I & A_i G_i - F_i \\
G_i^T A_i^T - F_i^T & P_j - G_i - G_i^T
\end{bmatrix} < 0,
\quad \forall (i,j) \in I \times I,
$$

(5.25)

Note that inequalities (5.25) represent strict LMIs but the constraints (5.22) and (5.23) are non-strict. Recall that minimization under non-strict LMI constraints gives the same result as minimization under strict LMI constraints when both strict and non-strict LMI constraints are feasible [24]. This is the case for (5.25).

To obtain a well-posed optimization problem, we should normalize $\rho$ to 1 since the ratio $\nu/\rho$ can be made arbitrarily small by choosing sufficiently large $\rho$ without violating constraints (5.25). With

$$\begin{align*}
1 & \leftarrow \rho, \\
\nu & \leftarrow \nu/\rho, \\
\eta & \leftarrow \eta/\rho, \\
G_i & \leftarrow G_i/\rho, \\
P_i & \leftarrow P_i/\rho,
\end{align*}$$

the following optimization problem is proposed:

maximize $\nu$
subject to
\[ \eta I < P_i < I, \]
\[ \left[ \begin{array}{cc}
A_iF_i^T + F_iA_i^T - P_i + \nu I & A_iG_i - F_i \\
G_i^TA_i^T - F_i^T & P_j - G_i - G_i^T
\end{array} \right] < 0, \]
\[ \nu, \eta > 0, \forall (i, j) \in I \times I, \] (5.26)

**Remark 5.6** Constraints \( \eta I < P_i < I \) \((i \in I)\) limit the condition number of \( P_i \) to \( 1/\eta \). The advantage of the condition number limit is that it will prevent the LMI solution algorithm from converging to \( P_i \) that could lead to roundoff problems.

The following theorem answers the Q2):

**Theorem 5.6** Switched system (5.12) is exponentially stable with a decay rate \( \xi = (1 - \nu)^{1/2} \) if the problem (5.26) is feasible. The assured bound on the decay rate is given by \( (1 - \nu_{opt})^{1/2} \) with \( \nu_{opt} \) the optimal value of optimization problem (5.26).

**Remark 5.7** It is possible to produce a better estimate of the decay rate of the switched system. However, a finer partitioning is needed [170]. A SLF having the same switching signals as the switched system may not be sufficient.

Switched State Feedback

An important aspect of the new conditions (5.17) given in Theorem 5.4 is that they are LMIs in \( P_i, F_i \) and \( G_i \) where there is no cross product between the matrix \( A_i \) and the Lyapunov matrix \( P_i \) \((i \in I)\). This fact has an impact on the synthesis problem considered below.

Let us consider the switched systems
\[ x(t+1) = A_\sigma x(t) + B_\sigma u(t) \] (5.27)
where \( u(t) \) is the control and the switching signal is available in real-time. The stabilizing state feedback control problem is to find
\[ u(t) = K_\sigma x(t) \] (5.28)
such that the corresponding closed-loop switched system (5.29) is stable.

\[ x(t + 1) = (A_{\sigma} + B_{\sigma}K_{\sigma})x(t) \quad (5.29) \]

The following theorem gives a sufficient condition to build a switched state feedback controller, which ensures the exponential stability of the closed-loop switched system. Moreover, this controller is optimal in the sense that the bound on the decay rate of the system is attained. Q3) is thus solved.

**Theorem 5.7** If there is a solution to

maximize \( \nu \)

subject to

\[
\eta I < P_i < I, \\
\begin{bmatrix}
-P_i + \nu I & A_i G_i + B_i R_i \\
G_i^T A_i^T + R_i^T B_i^T & P_j - G_i - G_i^T
\end{bmatrix} < 0,
\]

\( \nu, \eta > 0, \forall (i, j) \in \mathcal{I} \times \mathcal{I}, \)

then the state feedback control given by (5.28) with

\[ K_i = R_i G_i^{-1}, \forall i \in \mathcal{I} \quad (5.30) \]

exponentially stabilizes the system (5.27). The decay rate of the system is given by \( \xi = (1 - \nu)^{\frac{1}{2}} \).

**Proof:** Conditions of Theorem 5.7 lead to

\[
\eta I < P_i < I, \\
\begin{bmatrix}
-P_i + \nu I & (A_i + B_i K_i) G_i \\
G_i^T (A_i + B_i K_i)^T & P_j - G_i - G_i^T
\end{bmatrix} < 0,
\]

\( \nu, \eta > 0, \forall (i, j) \in \mathcal{I} \times \mathcal{I}, \)
which are equivalent to condition (5.26) written for the closed-loop system (5.29) with \( F_i = 0 \). The result then follows from Theorem 5.6. Note that satisfying (5.30) implies \( P_j - G_i - G_i^T < 0 \) and matrices \( G_i \) are non-singular. Hence the following feedback gain \( K_i = R_i G_i^{-1} \) is always available whenever (5.30) are feasible. \( \square \)

**Remark 5.8** The condition given in Theorem 5.7 can also be adapted to the switched static output feedback control as shown in [36].

### 5.3.4 Input-Output Performance

The next step is naturally to consider the robustness of the switched state feedback control. This is the last question, Q4) proposed in Section 5.3.1. We study the robustness from a \( \gamma \)-performance point of view, that is the designed switched feedback control ensures that the worst case energy amplitude gain of the closed loop system is less than or equal to some specified positive level \( \gamma \).

\( \gamma \)-performance

Consider an autonomous discrete-time switched system given by

\[
\begin{align*}
    x(t+1) &= A_\sigma x(t) + B_\sigma^w w(t) \\
    z(t) &= C_\sigma^z x(t) + D_\sigma^z w(t)
\end{align*}
\]  \tag{5.31}

where \( x(t) \in \mathbb{R}^n \) is the state, \( x(0) = 0 \), \( w(t) \in \mathbb{R}^q \) is the disturbance and \( z(t) \in \mathbb{R}^p \) is an output vector. The switching rule \( \sigma \) is defined as previously. Similarly, the matrices \( (A_\sigma, B_\sigma^w, C_\sigma^z, D_\sigma^z) \) are allowed to take values, at an arbitrary time, in the finite set

\[
\{(A_1, B_1^w, C_1^z, D_1^z), \ldots, (A_N, B_N^w, C_N^z, D_N^z)\}.
\]

Given \( \gamma > 0 \), the \( \gamma \)-performance for the switched system (5.31) is defined as below.
Definition 5.3 ([35]) The autonomous system (5.31) is said to have a $\gamma$-performance if it is asymptotically stable and

$$\sum_{t=0}^{\infty} z^T(t)z(t) < \gamma^2 \sum_{t=0}^{\infty} w^T(t)w(t), \quad (5.32)$$

$$\forall w(t) \in \mathcal{L}_2, \ i.e., \ \sum_{t=0}^{\infty} w^T(t)w(t) < \infty.$$ 

Now, define the same Lyapunov function $V(t, x(t)) > 0, x \neq 0$ considered in Section 5.3.2, and the modified Lyapunov stability conditions

$$\Delta V(t, x(t)) < 0, \ \gamma^2 w^T(t)w(t) \leq z^T(t)z(t),$$

$$\forall (x(t), x(t+1), w(t), z(t)) \text{ satisfying } (5.31),$$

$$(x(t), x(t+1), w(t), z(t)) \neq 0,$$

for a given $\gamma$. The S-procedure [24] is invoked to generate the equivalent condition

$$\Delta V(t, x(t)) < \gamma^2 w^T(t)w(t) - z^T(t)z(t),$$

$$\forall (x(t), x(t+1), w(t), z(t)) \text{ satisfying } (5.31),$$

$$(x(t), x(t+1), w(t), z(t)) \neq 0.$$

If (5.34) is feasible for some $0 < \gamma < \infty$ then it is possible to conclude that the system (5.31) is internally asymptotically stable and has a $\gamma$-performance since

$$0 < V(t+1, x(t+1)) \leq \gamma^2 \sum_{k=0}^{t} w^T(k)w(k) - \sum_{k=0}^{t} z^T(k)z(k),$$

which is valid for all $t > 0$. In particular, take $t \to \infty$,

$$0 < V(\infty) < \gamma^2 \sum_{k=0}^{\infty} w^T(k)w(k) - \sum_{k=0}^{\infty} z^T(k)z(k), \quad (5.35)$$

which implies $\gamma^2 \sum_{k=0}^{\infty} w^T(k)w(k) > \sum_{k=0}^{\infty} z^T(k)z(k)$.

We are now ready to state the following theorem which gives a sufficient condition to check if the autonomous system (5.31) has a $\gamma$-performance.
Theorem 5.8 The system (5.31) has a $\gamma$-performance if there exist $P_i \in \mathbb{S}_+^n$, $F_{ii}, G_{ii} \in \mathbb{R}^{n \times n}$, $F_{2i}, G_{2i} \in \mathbb{R}^{n \times p}$, $H_{ii} \in \mathbb{R}^{p \times n}$, $J_{ii} \in \mathbb{R}^{q \times n}$, $H_{2i} \in \mathbb{R}^{p \times p}$, $J_{2i} \in \mathbb{R}^{q \times p}$, $i = \{1, \ldots, N\}$, such that

$$P + U + U^T < 0,$$

(5.36)

where

$$P = \begin{bmatrix} P_i & 0 & 0 & 0 \\ 0 & -P_j & 0 & 0 \\ 0 & 0 & I & 0 \\ 0 & 0 & 0 & -\gamma^2 I \end{bmatrix}, \quad \forall (i, j) \in (I \times I),$$

(5.37)

$$U = \begin{bmatrix} F_{ii}A_i + F_{2i}C_i^z & -F_{ii} & -F_{2i} & F_{1i}B_i^w + F_{2i}D_i^z \ 
G_{ii}A_i + G_{2i}C_i^z & -G_{ii} & -G_{2i} & G_{1i}B_i^w + G_{2i}D_i^z \\
H_{ii}A_i + H_{2i}C_i^z & -H_{ii} & -H_{2i} & H_{1i}B_i^w + H_{2i}D_i^z \\
J_{ii}A_i + J_{2i}C_i^z & -J_{ii} & -J_{2i} & J_{1i}B_i^w + J_{2i}D_i^z \end{bmatrix}.$$

Proof: Assign

$$x \leftarrow \begin{bmatrix} x(t) \\ x(t+1) \\ z(t) \\ w(t) \end{bmatrix}, \quad P \leftarrow \begin{bmatrix} -P_i & 0 & 0 & 0 \\ 0 & P_j & 0 & 0 \\ 0 & 0 & I & 0 \\ 0 & 0 & 0 & -\gamma^2 I \end{bmatrix},$$

$$H^T \leftarrow \begin{bmatrix} A_i^T & C_i^T \\ -I & 0 \\ 0 & -I \\ B_i^T & D_i^T \end{bmatrix}, \quad X \leftarrow \begin{bmatrix} F_{ii} & F_{2i} \\ G_{ii} & G_{2i} \\ H_{ii} & H_{2i} \\ J_{ii} & J_{2i} \end{bmatrix},$$

and apply Lemma 5.3 on (5.34). $\square$
Switched Feedback Control with $\gamma$-performance

In this section, we consider switched linear systems given by

$$\begin{cases}
  x(t+1) = A_\sigma x(t) + B_\sigma u(t) + B_\sigma^w w(t) \\
  z(t) = C_\sigma^z x(t) + D_\sigma^z u(t) + D_\sigma^zw(t)
\end{cases}$$  \hspace{1cm} (5.38)

A $\gamma$-performance state feedback problem is to find a control (5.28) such that the corresponding closed-loop switched system

$$\begin{cases}
  x(t+1) = (A_\sigma + B_\sigma K_\sigma)x(t) + B_\sigma^w w(t) \\
  z(t) = (C_\sigma^z + D_\sigma^z K_\sigma)x(t) + D_\sigma^zw(t)
\end{cases}$$  \hspace{1cm} (5.39)

has a $\gamma$-performance. The following theorem gives a sufficient condition to build such a controller.

**Theorem 5.9** There exists a switched state feedback control (5.28) such that the closed-loop switched system (5.39) has a $\gamma$-performance if there exist $P_i \in \mathbb{S}^n_+$, $G_{1i} \in \mathbb{R}^{n \times n}$, $F_{2i}$, $G_{2i} \in \mathbb{R}^{m \times n}$, $H_{2i} \in \mathbb{R}^{m \times m}$, $J_{2i} \in \mathbb{R}^{m \times q}$, $R_i \in \mathbb{R}^{m \times n}$, $i = \{1, \ldots, N\}$, such that

$$P + U + U^T < 0,$$  \hspace{1cm} (5.40)

where $P$ is given by (5.37) and

$$U = \begin{bmatrix}
  B_{1i}^w F_{2i} & A_i G_{1i} + B_i R_i + B_i^w G_{2i} & B_{1i}^w H_{2i} & B_{1i}^w J_{2i} \\
  0 & -G_{1i} & 0 & 0 \\
  -F_{2i} & -G_{2i} & -H_{2i} & -J_{2i} \\
  D_{i}^{zw} F_{2i} & C_i G_{1i} + D_i R_i + D_i^{zw} G_{2i} & D_i^{zw} H_{2i} & D_i^{zw} J_{2i}
\end{bmatrix}.$$

The $\gamma$-performance state feedback control law is given by (5.28) with $K_i = R_i G_{1i}^{-1}$.

**Proof:** Use a transposed version of Theorem 5.8 with

$$F_{1i} = 0, \quad H_{1i} = 0, \quad J_{1i} = 0,$$  \hspace{1cm} (5.41)
and \( R_i = K_i G_{1i} \).

In order to obtain a convex condition (5.40), we made the choice in (5.41) but it is not unique. There are other choices that can also lead to a convex condition.

**Remark 5.9** By setting \( F_{2i} = 0, G_{2i} = 0, H_{2i} = I \), and \( J_{2i} = 0 \), we can recover the corresponding condition in [35]. Without restrictions on \( F_{2i}, G_{2i}, H_{2i}, \) and \( J_{2i} \), a better \( \gamma \)-performance level with respect to the one in [35] can be obtained. However, this came at expense of a more intensive computation; see also [45].

**Remark 5.10** The best upper bound on the \( L_2 \)-induced gain can be achieved by minimizing \( \gamma \) subject to the constraints defined by (5.40), which is a classical eigenvalue problem [24].

Appendix I: Proof of Lemma 5.1

The left inclusion in (5.5) is a direct consequence of the inclusion \( G(x) \subseteq \text{co}G(x) \). In the following, the right inclusion is proved by induction.

At time \( k = k_0 \) and \( k = k_0 + 1 \), the inclusion holds automatically, i.e., \( Y(k_0, k_0, x_0) = X(k_0, k_0, x_0) \equiv x_0 \), and \( X(k_0 + 1, k_0, x_0) = G(x_0), Y(k_0 + 1, k_0, x_0) = \text{co}G(x_0) \). Assume now that the right inclusion in (5.5) is satisfied at some time \( k = \bar{k} > k_0 \), i.e.,

\[
Y(\bar{k}, k_0, x_0) \subseteq \text{co}X(\bar{k}, k_0, x_0).
\]  

Let us prove the inclusion is also satisfied at time \( k = \bar{k} + 1 \). By definition,

\[
X(\bar{k} + 1, k_0, x_0) = G(X(\bar{k}, k_0, x_0)) = \bigcup_{x \in X(\bar{k}, k_0, x_0)} G(x),
\]

\[
Y(\bar{k} + 1, k_0, x_0) = \text{co}G(Y(\bar{k}, k_0, x_0)) = \bigcup_{y \in Y(\bar{k}, k_0, x_0)} \text{co}G(y).
\]
Since
\[ \bigcup_{y \in Y(k,k_0,x_0)} \text{co}G(y) \subseteq \text{co} \left( \bigcup_{y \in Y(k,k_0,x_0)} G(y) \right), \]
and by virtue of (5.42) and (5.43) we have
\[ Y(\bar{k} + 1, k_0, x_0) \subseteq \text{co} \left( \bigcup_{x \in \text{co}X(k,k_0,x_0)} G(x) \right) = \text{co}(G(\text{co}X(\bar{k}, k_0, x_0))). \tag{5.44} \]

Note that for any set \( X \subset \mathbb{R}^n \) it holds that
\[ \text{co}(G(\text{co}X)) = \text{co}G(X). \tag{5.45} \]

In fact, \( \text{co}G(X) \subseteq \text{co}(G(\text{co}X)) \) follows from inclusions \( x \subseteq \text{co}X \) and \( G(x) \subseteq G(\text{co}X) \). Now consider any point \( x \in \text{co}(G(\text{co}X)) \). By Caratheodory’s Theorem, there exist \( \bar{x}_i \in G(\text{co}X), i = 1, \cdots, n+1 \) such that
\[ x = \sum_{i=1}^{n+1} \lambda_i \bar{x}_i \tag{5.46} \]
for some nonnegative \( \lambda_i \) with \( \sum_{i=1}^{n+1} \lambda_i = 1 \). On the other hand, for any point \( \bar{x}_i \in G(\text{co}X) \) \( (i = 1, \cdots, n+1) \), there exists matrix \( A_i \in \mathcal{A} \), such that
\[ \bar{x}_i = A_i \sum_{j=1}^{n+1} \mu_{ji} x_{ji} \tag{5.47} \]
for some nonnegative \( \mu_{ji} \) with \( \sum_{j=1}^{n+1} \mu_{ji} = 1 \) and \( x_{ji} \in X, i,j = 1, \cdots, n+1 \).

Substitute (5.47) in (5.46), we get
\[ x = \sum_{j=1}^{n+1} \sum_{i=1}^{n+1} \mu_{ji} \lambda_i A_i x_{ji} = \sum_{i,j=1}^{n+1} \gamma_{ji} \hat{x}_{ji}, \tag{5.48} \]
where \( \gamma_{ji} = \mu_{ji} \lambda_i \geq 0 \) with
\[ \sum_{i,j=1}^{n+1} \gamma_{ji} = \sum_{i,j=1}^{n+1} \mu_{ji} \lambda_i = \sum_{i=1}^{n+1} \lambda_i \sum_{j=1}^{n+1} \mu_{ji} = \sum_{i=1}^{n+1} \lambda_i = 1, \]
and \( \hat{x}_{ji} = A_i x_{ji} \in G(x), i, j = 1, \ldots, n + 1. \)

From (5.48) it follows that \( x \in \text{co}G(X) \) and the inclusion \( \text{co}G(X) \subseteq \text{co}(G(\text{co}X)) \) is established. Together with the inclusion \( \text{co}(G(X)) \subseteq \text{co}G(X) \), we prove (5.45).

From (5.44) and (5.45), we have

\[
Y(\bar{k} + 1, k_0, x_0) \subseteq \text{co}(G(X(\bar{k}, k_0, x_0))),
\]

and from (5.43) we obtain the inclusion

\[
Y(\bar{k} + 1, k_0, x_0) \subseteq \text{co}X(\bar{k} + 1, k_0, x_0)
\]

That finishes the induction and proves the right inclusion in (5.5). This thus completes the proof of Lemma 5.1. The lemma thus follows.

Appendix II: Lie Algebras

A Lie algebra \( \mathfrak{g} \) is a finite-dimensional vector space equipped with a Lie bracket, i.e., a bilinear, skew-symmetric map \([\cdot, \cdot] : \mathfrak{g} \times \mathfrak{g} \to \mathfrak{g}\) satisfying the Jacobi identity

\[
[a, [b, c]] + [b, [c, a]] + [c, [a, b]] = 0.
\]

Any Lie algebra \( \mathfrak{g} \) can be identified with a tangent space at the identity of a Lie group \( G \) (an analytic manifold with a group structure).

If \( \mathfrak{g} \) is a matrix Lie algebra, then the elements of \( G \) are given by products of the exponentials of the matrices from \( \mathfrak{g} \). In particular, each element \( A \in \mathfrak{g} \) generates the one-parameter subgroup \( \{e^{At}, t \in \mathbb{R}\} \) in \( G \). For example, if \( \mathfrak{g} \) is the Lie algebra \( gl(n, \mathbb{R}) \) of all real \( n \times n \) matrices with the standard Lie bracket \([A, B] = AB - BA\), then the corresponding Lie group is given by the invertible matrices.

Given an abstract Lie algebra \( \mathfrak{g} \), one can consider its (matrix) representations. A representation of \( \mathfrak{g} \) on an \( n \)-dimensional vector space \( V \) is a homomorphism (i.e., a linear map that preserves the Lie bracket) \( \phi : \mathfrak{g} \to gl(V) \). It assigns to each element \( g \in \mathfrak{g} \) a linear operator \( \phi(g) \) on \( V \), which can be described by an \( n \times n \) matrix. A useful representation is the adjoint one, denoted by “ad.” The vector space \( V \) in this
case is $\mathfrak{g}$ itself, and for $g \in \mathfrak{g}$ the operator $\text{ad}g$ is defined by $\text{ad}g(a) := [g, a], \ a \in \mathfrak{g}$.

There is also Ado’s Theorem which says that every Lie algebra is isomorphic to a subalgebra of $\mathfrak{gl}(V)$ for some finite-dimensional vector space $V$. 
6.1 A Summary of Contributions

In this chapter, we review the contributions of this dissertation and indicate several future research directions.

In Chapter 2, we cast existing work on consensus protocols into an asynchronous framework. This extension lends powerful results to the consensus problem. The model of asynchronous multi-agent systems encompasses as a special case the synchronous one with various communication patterns, i.e., issues of directional, delayed, or failed communication. They can thus be addressed in the same framework.

Chapter 3 examined the different assumptions made in the various consensus results in the literature so to better understand their roles in the convergence analysis of consensus protocols. For example, the differences between unidirectional communication and bidirectional communication were made clear from a matrix operator point of view. A novel nonlinear asynchronous consensus result was also introduced using the theory of paracontractions and confluence. This result is more general than the existing ones and provides a powerful tool to study a wider range of applications.

In Chapter 4, we developed a basic formation tracking controller (FTC) as well as a consensus-based one using only neighboring leaders information. The stability properties of the multiple vehicle system in closed loop with these FTCs were studied
using cascaded systems theory and nonlinear synchronization theory. In particular, we established connections between stability of consensus-based FTC and Laplacian matrices for formation graphs. The global results obtained allow us to consider a large class of trajectories with arbitrary (rigid) formation patterns and initial conditions. Our formation tracking strategy holds great potential to be extended to the case of air and marine vehicles.

In Chapter 5, we explored the stability equivalence between linear time-varying systems and switched linear systems. A new proof is provided based on Caratheodary’s Theorem in convex analysis. Only discrete-time systems are investigated but most of the results presented can be extended to the continuous-time case [105]. The switched Lyapunov function method has been combined with Finsler’s Lemma to study switched linear systems. New and less conservative LMI conditions are developed for stability and control design problems with performance analysis. Output feedback control problem can also be treated in a similar way by using the technique developed in this chapter.

6.2 Future Research Directions

Here, we outline research directions which spring from this dissertation.

**Asynchronous consensus problem**: Many open problems still exist; see [54] for a detailed discussion. In view of the results of Example 2.2 it is important to understand how topologies, delays and updating sets affect the value of the consensus point. Another remark concerns the asymptotic property of the asynchronous consensus protocol. In some cases, it is desirable for the robots to converge to a single point within a finite time instead of infinite time for asymptotic protocols. Therefore, the finite convergent property of consensus protocols deserves further study. In addition, it will be interesting to use optimization tools such as stochas-
tic approximation [187] to study the asynchronous consensus problem in stochastic environments.

**Formation tracking control:** We did not discuss collision avoidance and formation error propagation problems. Our formation tracking controller does not guarantee avoidance of collisions and there is a need to take care of them in the future work. Furthermore, Corollary 4.1 showed that two vehicles with a cascaded interconnection is formation ISS. Its invariance properties under cascading [113] could be explored to quantify the formation errors when individual vehicle’s tracking errors are bounded.

**Maximizing the graph connectivity:** The algebraic connectivity of a graph is the second smallest eigenvalue of the graph Laplacian (denoted as $\lambda_2(L_G)$), and is a measure of how well-connected the graph is. In [131] the parameter $\lambda_2(L_G)$ has been shown to play a crucial role in the network consensus problem. For a graph with a fixed topology, its algebraic connectivity can be maximized by adjusting the weights of Laplacian matrix [200]. Another way to improve the algebraic connectivity of a graph is to add a set of candidate edges to the graph. This is a difficult combinatorial optimization. We may utilize matrix perturbation analysis [184] and rounding via random projection method [189] to solve the problem.
BIBLIOGRAPHY


[174] Y. Su, A. Bhaya, E. Kaszkurewicz, and V. S. Kozyakin. Further results on

[175] Yangfeng Su and Amit Bhaya. Convergence of pseudocontractions and appli-


[178] D. B. Szyld. The mystery of asynchronous iterations convergence when the

[179] H. G. Tanner, G. J. Pappas, and V. Kumar. Input-to-state stability on for-


[181] D. Teneketzis and P. Varaiya. Consensus in distributed estimation with in-

[182] H. T. Toivonen and P. M. Makila. A descent anderson-moore algorithm for


[184] M. Torki. Second-order directional derivatives of all eigenvalues of a symmet-
ric matrix. *NONLINEAR ANALYSIS-THEORY METHODS & APPLICA-

[185] J. N. Tsitsiklis and M. Athans. Convergence and asymptotic agreement in
1984.

[186] J. N. Tsitsiklis, D. P. Bertsekas, and M. Athans. Distributed asyn-
chronous deterministic and stochastic gradient optimization algorithms. *IEEE

[187] F. J. Vazquez-Abad, C. G. Cassandras, and V. Julka. Centralized and decen-
tralized asynchronous optimization of stochastic discrete-event systems. *IEEE


