

# Coordinated Control for Networked Multi-Agent Systems

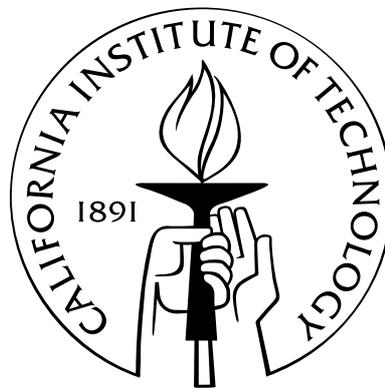
Thesis by

Zhipu Jin

In Partial Fulfillment of the Requirements

for the Degree of

Doctor of Philosophy



California Institute of Technology

Pasadena, California

2007

(Defended September 1, 2006)



To my parents, my brother, and my dear wife

# Acknowledgements

First of all, I would like to express my heartfelt gratitude to my advisor, Prof. Richard M. Murray, for his guidance that is so conducive to the work I have undertaken. His broad knowledge, deep insights, outstanding leadership, and great personality make him a mentor that every student dreams of having. I am grateful to my committee members: Prof. John C. Doyle, Prof. Joel W. Burdick, Prof. Babak Hassibi, and Prof. Tracey Ho, for inspiring discussions and detailed comments on my thesis. I also appreciate the help from Prof. Michelle Effros, Prof. Leonard Schulman and Prof. Steven Low.

There are many other people I would like to acknowledge with thankfulness for the help given to me in the last five years. Thanks to Dr. Alcherio Martinoli for the unforgettable support during my first year at Caltech. Thanks to Dr. Reza Olfati-Saber for showing me his wonderful work on coordinated control. Thanks to Dr. Eric Klavins, Dr. Lars Cremean, David van Gogh, and Steve Waydo for helping me conduct experiments on the testbed. Also, thanks to Vijay Gupta, Ling Shi, Abhishek Tiwari, and other group members for collaborations and brainstorming. Because of them, my research became much easier and more fruitful. Also thanks to staff members at Caltech for making my administrative life easy.

I would like to thank Lijun Chen, Xiaoli Feng, Hao Jiang, Lun Li, Yongqiang Liang, Xin Liu, Changlin Pang, Zhengrong Wang, Qiang Yang, Chengzhong Zhang, Kechun Zhang, Yizhen Zhang, and all other friends I made here. Thanks to them for bringing me so many enjoyable memories.

Last but not least, I profoundly thank my family. Their constant understanding and support is the source of all my strength and courage. I owe too much to my lovely wife, Yi, and I do not think it is possible that I could have come this far without her love.

# Abstract

Coordination in networked multi-agent systems attracts significant interest in the realm of engineering. Typical examples include formations of unmanned aerial vehicles, automated highway systems, and sensor networks. One common feature for these systems is that coordinated behaviors are exhibited by interactions among agents where information exchange and manipulation are necessary. In this work, three relevant issues are investigated in detail: uniform strategy for multi-agent formation control, fast-converging consensus protocols, and packet-based state estimation over communication networks.

Formation control of multi-agent systems involves harmony among local controller design, interaction topology analysis, and objective agreement among networked agents. We propose a novel control strategy so that each agent responds to neighbors' behaviors as well as acts towards the global goal. Thus, information flows for local interactions and global objective synchronization are studied separately. Using the tools from signal flow graphs and algebraic graph theory, we show that this new strategy eases the design of local controllers by relaxing stabilizing conditions. Robustness against the link failure and scalable disturbance resistance are also discussed based on small-gain theory. Experimental results on the Caltech multi-vehicle wireless testbed are provided to verify the feasibility and efficiency of this control strategy.

Consensus protocols over communication networks are used to achieve agreement among agents. One important issue is the convergence speed. We propose multi-hop relay protocols for fast consensus seeking. Without physically changing the topology of the communication network, this type of distributed protocol increases the algebraic connectivity by employing multi-hop paths in the network. We also investigate the convergence behaviors of consensus protocols with communication delays. It is interesting that, the faster the pro-

protocol converges, the more sensitive it is to the delay. This tradeoff is identified when we investigate delay margins of multi-hop relay protocols using the frequency sweep method.

Efficiently estimating the states of other agents over communication links is also discussed in this work. When information flows in the network, packet-based data is normally not retransmitted in order to satisfy real-time requirements. Thus, packet drops and random delays are inevitable. In this context, observation data that the estimator can receive is intermittent. In order to decrease the chance of losing packets and efficiently using the limited bandwidth, we introduce multiple description source codes to manipulate the data before transmission. Using modified algebraic Riccati equations, we show that multiple description codes improve the performance of Kalman filters over a large set of packet-dropping scenarios. This problem is also generalized to the case where observation data has an independent and identical static distribution over a finite set of observation noise. Moreover, Kalman filtering with bursty packet drops is also discussed based on the two-state Markov chain model.

# Contents

<b>Acknowledgements</b>	<b>iv</b>
<b>Abstract</b>	<b>v</b>
<b>1 Introduction</b>	<b>1</b>
1.1 Background and Motivation . . . . .	1
1.2 Previous Work . . . . .	4
1.2.1 Multi-Agent Formation Control . . . . .	4
1.2.2 Collective Behaviors and Consensus Seeking . . . . .	7
1.2.3 Control and Estimation over Networks . . . . .	8
1.3 Statement of Contributions . . . . .	11
<b>2 Preliminaries</b>	<b>13</b>
2.1 Algebraic Graph Theory . . . . .	14
2.1.1 Basic Concepts . . . . .	14
2.1.2 Matrices in Algebraic Graph Theory . . . . .	15
2.1.3 Infinity Norm of Normalized Adjacency Matrix . . . . .	21
2.2 Quantization and Distortion . . . . .	24
2.3 Modeling for Packet Drops . . . . .	26
<b>3 Double-Graph Control Strategy for Formation Control</b>	<b>29</b>
3.1 Formulation of Double-Graph Control Strategy . . . . .	29
3.1.1 Gain Matrix in Signal Flow Graphs . . . . .	29
3.1.2 Nyquist Criterion for Formation Stability . . . . .	32

3.1.3	Double-Graph Control Strategy . . . . .	35
3.2	Stability Analysis for Double-Graph Control Strategy . . . . .	38
3.3	Performance for Double-Graph Control Strategy . . . . .	43
3.3.1	Disturbance Resistance with Acyclic Interaction Topologies . . . . .	44
3.3.2	Disturbance Resistance with Arbitrary Interaction Topologies . . . . .	45
3.3.3	Other Performance Issues . . . . .	47
3.4	Simulation and Experimental Results . . . . .	48
3.4.1	Multi-Vehicle Wireless Testbed . . . . .	48
3.4.2	Simulation Results . . . . .	50
3.4.3	Experimental Results . . . . .	52
<b>4</b>	<b>Consensus Protocols in Networked Multi-Agent Systems</b>	<b>57</b>
4.1	Consensus Seeking for Multi-Agent Systems . . . . .	57
4.2	Multi-Hop Relay Protocols for Consensus Seeking . . . . .	63
4.2.1	Two-Hop Relay Protocol . . . . .	64
4.2.2	Convergence Speed of Two-Hop Relay Protocol . . . . .	66
4.2.3	Multi-Hop Relay Protocol . . . . .	67
4.3	Two-Hop Relay Protocols with Time Delays . . . . .	67
4.4	Examples and Simulation Results . . . . .	73
<b>5</b>	<b>Packet-Based State Estimation Using Multiple Description Codes</b>	<b>78</b>
5.1	Formulation and Assumptions for Packet-Based State Estimation . . . . .	79
5.2	Multiple Description Source Codes . . . . .	81
5.2.1	Theoretical Limits of Multiple Description Codes . . . . .	83
5.2.2	Multiple Description Scalar Quantizer . . . . .	84
5.2.3	Quantization Noise of MD codes . . . . .	86
5.3	Kalman Filtering Utilizing MD Codes . . . . .	87
5.3.1	Kalman Filtering with i.i.d. Packet Drops . . . . .	87
5.3.2	Discussion for Generalized MARE . . . . .	98
5.3.3	Kalman Filtering with Bursty Packet Drops . . . . .	100
5.4	Simulation Results . . . . .	105

<b>6</b>	<b>Conclusions and Future Work</b>	<b>110</b>
6.1	Summary of Main Contributions . . . . .	110
6.2	Future Directions and Possible Extensions . . . . .	112
	<b>Bibliography</b>	<b>115</b>
<b>A</b>	<b>Index Assignment Method for MDSQ</b>	<b>129</b>
<b>B</b>	<b>Additional Proofs</b>	<b>132</b>

# List of Figures

1.1	Architectures for centralized, decentralized, and distributed control architectures . . . . .	1
1.2	Diagram of a typical networked control system . . . . .	9
2.1	Classification for directed graphs . . . . .	15
2.2	An example: directed graph with seven vertices . . . . .	17
2.3	Reduction of a directed graph . . . . .	20
2.4	Bounds of normalized adjacency matrix norm . . . . .	24
2.5	Models of packet drops: (a) i.i.d. Bernoulli model; (b) two-state Markov chain model . . . . .	27
3.1	A signal flow graph . . . . .	29
3.2	Interaction topology and the corresponding signal flow graph . . . . .	34
3.3	Double-graph control strategy for multi-agent systems . . . . .	37
3.4	Diagram of the local control strategy . . . . .	38
3.5	Interaction topology of a leader-follower system with twelve agents . . . . .	39
3.6	Nyquist plot and critical points . . . . .	40
3.7	Mapping for critical points . . . . .	41
3.8	Ellipses for critical points with different $\alpha$ . . . . .	42
3.9	Nyquist plot and critical points for double-graph control strategy . . . . .	42
3.10	Disturbance for single agent . . . . .	44
3.11	Schematic plot of MVWT vehicle. . . . .	49
3.12	Control diagram for lateral dynamics. . . . .	51
3.13	Three interaction topologies of vehicle formation . . . . .	51

3.14	Top view of a single vehicle experiment . . . . .	53
3.15	Lateral errors for vehicle formation with different parameters . . . . .	54
3.16	Topologies in double-graph control strategy of the MVWT experiment . . . . .	54
3.17	Top view of vehicle formation experiment . . . . .	55
3.18	Vehicle formation experiment: radius and speed . . . . .	56
4.1	A directed graph and its two-hop directed graph . . . . .	64
4.2	An example of disconnected two-hop directed graph . . . . .	67
4.3	Locus of the zero of the quasipolynomial and the generalized eigenvalue . . . . .	71
4.4	Three different topologies: $\mathcal{G}_1$ , $\mathcal{G}_2$ , and $\mathcal{G}_3$ . . . . .	73
4.5	States of graph $\mathcal{G}_2$ with no delay . . . . .	75
4.6	States of graph $\mathcal{G}_2$ with delay $\tau = 0.038$ . . . . .	75
4.7	States of graph $\mathcal{G}_2$ with delay $\tau = 0.05$ . . . . .	76
4.8	States of graph $\mathcal{G}_2$ with delay $\tau = 0.25$ . . . . .	76
4.9	Tradeoff between $\lambda_2$ and $\tau^*$ . . . . .	77
5.1	Diagram of packet-based state estimation . . . . .	80
5.2	Diagram of two-description MD source encoder . . . . .	85
5.3	Additive noise model of uniform scalar quantization . . . . .	86
5.4	Simulation results of expected error covariances with theoretical upper and lower bounds . . . . .	105
5.5	Mean values of error covariance with same central distortions . . . . .	106
5.6	Mean values of error covariance with same bpss . . . . .	107
5.7	Mean values of error covariance with low dropping rate . . . . .	107
5.8	Theoretical upper and lower bounds for burst packet-dropping case with $q_{11} = 95\%$ . . . . .	108
5.9	Simulation results for burst packet-dropping case . . . . .	109
A.1	Two actions to fill the index mapping matrix . . . . .	130
A.2	Examples of index mapping . . . . .	130

## List of Tables

4.1	Possible left eigenvectors for $\mathcal{L}$ associated with $\lambda(\mathcal{L}) = 0$ . . . . .	58
4.2	Performance vs. robustness for relay protocols . . . . .	74
5.1	Average distortion for different MD codes . . . . .	86

# Chapter 1

## Introduction

### 1.1 Background and Motivation

As an engineering discipline, automatic control theory is always tied to practical problems encountered in human history. Classical control theory relies on transform methods in the frequency domain to deal with single-input-single-output (SISO) systems. Starting in 1960's, the state-space method has been used for multiple-input-multiple-output (MIMO) systems. Efficient algorithms for solving matrix equations and advanced microprocessors for data processing popularize this time-domain approach. However, for large scale control objects, such as electric power grids or large oil refineries, thousands of signals and variables need to be sensed and calculated. The lack of full sensor information and limited computation capability make the conventional centralized methods fail.

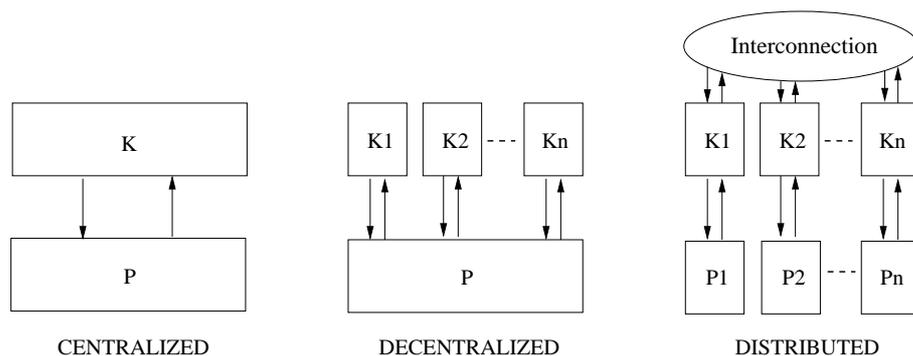


Figure 1.1: Architectures for centralized, decentralized, and distributed control architectures

This crisis was realized by the systems and control community in the 1970's and numer-

ous efforts has been made [1]. Three different control architectures for large scale systems are shown in Figure 1.1. For the first architecture, a centralized controller  $K$  is used to control a dynamic system  $P$ . One direct way to solve the complexity crisis is model reduction (or model simplification) [2, 3, 4]. This approach tries to reduce the order of state space by simplifying the system structure so that the complexity of the central controller can be moderated.

The second architecture in Figure 1.1 decentralizes the feedback controller. Instead of using one centralized controller, which requires full (or controllable) state feedback, spatially separated controllers  $\{K_1, K_2, \dots, K_n\}$  are introduced. Generally, each controller only accesses part of the system states and there are no information exchanges among the controllers. The most definitive results on decentralized stabilization can be found in [5, 6]. Note that the term “decentralization” refers to controller implementation and the control law is designed in a centralized mode.

The third architecture is a distributed control architecture where control object is composed of multiple decoupled subsystems  $\{P_1, P_2, \dots, P_n\}$ . Each subsystem is equipped with a local controller. Interconnection exists among these controllers so that information can be exchanged and subsystems can interact with each other. Thus, distributed control laws are generated according to not only the local feedback, but also the messages from other controllers. Cooperative and coordinated control for networked multi-agent systems belongs to this structure.

If we use ordinary differential equations (ODEs) to describe dynamical systems, qualitative analysis for large scale systems can be studied under the framework of interconnected systems. A huge amount of related work has been reported in the literature. By assuming that each subsystem is stable when isolated, [7] and [8] summarize the early research and provide details on how to use the vector Lyapunov method and weighted-sum method to test the stability of interconnected systems. The interpretation of their results is that if the gain of the local feedback loop is smaller than 1 and the coupling between subsystems is weak enough, then the overall system is stable. These tests are only successful when the coupling among subsystems is weak. Moreover, the concept of input-output stability for interconnected systems is proposed in [9] and several exciting results are presented in

[10, 11, 12].

During the last few decades, cooperative and coordinated control for multi-agent systems has received significant attention. Dramatic developments in communication and computation technology relax the restrictions on information exchanges between spatially distributed agents. The possibility has been identified that simple and inexpensive agents can carry out complicated tasks through collaboration that may be too difficult for single agent. Advantages include simple structure, low cost, enhanced functionality, and the flexibility in fault tolerance. Applications of cooperative and coordinated multi-agent systems can be found in space exploration [13, 14, 15, 16, 17, 18, 19], automated highway systems [20, 21, 22, 23, 24, 25], autonomous combat systems [26, 27, 28], oceanographic sampling [29, 30, 31], air traffic control [32, 33], congestion control on the internet [34], and sensor networks [35, 36, 37]. An agent can represent a spaceship, an airplane, a ground/underwater vehicle, an internet router, a cellular phone, or even a smart sensor with microprocessors. Among these applications, a couple of issues have not yet been well answered:

- Interactions among agents make the dynamics of multi-agent systems more complicated than single agent. Practical design and implementation needs more constructive insights from qualitative analysis on stability and performance issues with respect to nontrivial agent dynamics and interaction topology.
- Agents are physically decoupled. But their behaviors are coupled through a certain task that they try to accomplish. Task decomposition and assignment are traditionally solved in a centralized manner. Their counterparts in distributed systems, which can avoid the complexity crisis, are still not fully addressed.
- Data links between different agents are not perfect. Even though current communication networks provide affordable platforms for information exchange, limited bandwidth, random transmission delay/packet drops, and uncertainty in connectivity topology violate conventional assumptions in automatic control theory. How this questionable fidelity impacts control theory is not well understood yet.

In order to conquer these challenges, developing new tools and techniques from the interdisciplinary territory of control, computation, and communications is necessary [38]. A short survey of recent related work is provided in the next section, which obviously is bounded by author's ignorance and biases. But, it is clear that control theory can definitely benefit from other closely related disciplines, such as graph theory, distributed computation, information theory, and networking analysis. This work is dedicated to presenting our recent work on coordinated control for networked multi-agent systems, which can be roughly divided into three topics: coordinated control strategy for networked multi-agent systems, multi-hop relay protocols that boost the process of consensus seeking, and packet-based state estimation over lossy communication networks.

## 1.2 Previous Work

### 1.2.1 Multi-Agent Formation Control

Formation control is a popular topic for multi-agent systems. The control objective is letting agents maintain certain geometric formation by autonomously responding to other agents and the environment. Formation control has been extensively investigated in numerous applications such as coordination of multiple robots [39, 40, 41, 42, 43], control of ground vehicle platoons [22, 23, 25], formation flight of unmanned aerial vehicles (UAVs) [26, 27, 28, 44], fleets of autonomous underwater vehicles (AUVs) [30, 31, 45], and satellite clusters [15, 16, 19]. Even though each application shows unique characteristics and challenges, there exist common features. In most of the applications, agents have identical dynamics and similar local controller structure. Also, communication and computation capacity for each agent is limited. Last, formation configurations are consistent with the third architecture in Figure 1.1, and the interaction topology plays an important role.

Various approaches for formation control have been proposed and they can be roughly categorized as the leader-follower approach, the virtual structure approach, and the behavior-based approach.

- In the leader-follower approach, one agent acts as the leader and other agents are

followers who try to follow the leader's behavior by reacting to their nearest neighbors. This approach is naturally implemented in a distributed pattern since each agent only needs to react to its local environment. However, there exist a couple of shortcomings. For example, the approach heavily depends on the leader status and the whole formation will fail if the leader fails. Also, the stability of single agent does not necessarily indicate the stability of the formation [46]. Thus, more critical stable conditions have to be posed on the local controller. Moreover, nontrivial agent dynamics results in disturbance accumulation, which is unscalable with respect to the number of agents [23].

- The virtual structure approach is commonly used in the robotics community [47, 39, 45, 48, 49]. The unique feature is that a “virtual” agent is synthesized based on all agents. The formation is treated as a rigid body and this fictitious agent acts as the reference. The position and trajectory of each other agent is calculated explicitly based on the reference and formation configuration. It is easy to describe the whole group and maintain accurate formation. However, this approach is only practical for small groups because centralized data collection and processing are needed. Recent attempts on distributed implementation are reported in [18, 50].
- In the behavior-based approach, the control law of each agent is defined by a combination of pre-defined control actions corresponding to all possible agent status. Even though this approach is distributed, it is difficult to analyze quantitatively. Limited applications for this approach are reported in [40, 51].

Obviously, formation control belongs to the distributed architecture. For agent  $P_i$ , we can present its dynamics as

$$\dot{x}_i = f(x_i, u_i(x_i, x_{\text{neighbor}})), \quad (1.1)$$

where  $x_{\text{neighbor}}$  is the information collected from the neighbors and determined by the interaction topology. The local controller  $K_i$  generates the control law  $u_i(x_i, x_{\text{neighbor}})$ . When agents have identical dynamics and local controllers, the interaction topology is the key for formation control. One powerful tool to study the interaction topology is graph theory [52],

a mathematic theory on the properties of graphs. The objects of graph theory are a set of vertices and the edges between them. The idea of using graphs to interpret the interaction topologies can be found as early as in [53], where the input-output stability is decomposed into the stability of a hierarchy of strongly connected subsystems. For formation control, each agent can be abstracted as a vertex, and edges are used to present the existence of interactions. In [39, 40], graphs are used for the virtual structure of mobile robots. In [41], a triplet of leader robot, formation shape variable, and control graph is introduced for path planning and formation control of nonholonomic robots. Specific graphs, such as strings, are used in coordinated control for vehicle platoons [22]. In these literatures, stability of the formation is straightforward either because of the trivial agent dynamics or the simplicity of the graph.

For interaction topologies with loops, formation stability becomes tricky. A branch of graph theory, algebraic graph theory, is especially beneficial because of its fruitful results on graphs in connection with linear algebra. In [54, 55], spectrum analysis of the Laplacian matrix, a subject in algebraic graph theory, is shown to be important for the synchronization of coupled oscillators. Lyapunov stability analysis in terms of the eigenvalues of the Laplacian matrix is reported in [56] for nonlinear systems. A more precise statement on formation stability with linear dynamics is presented in [46]. It has been shown that the formation stability is equivalent to the stability of a sequence of decoupled inhomogeneous subsystems.

Another issue for formation control is the disturbance accumulation phenomenon. A disturbance signal can be amplified when it propagates to other agents through interactions. This subject is identified as the “string stability” problem in [22, 23] where formation control of vehicle platoons is discussed. In order to bound this amplification, numerous designs for interaction topology are proposed. The results are extended to vehicle formation with mesh topologies [25]. The point is to weaken the coupling among vehicles so that the gain of the disturbance is less than 1.

Interactions in multi-agent systems can be explained by local information flows among agents. On-board sensing and wireless communication links are common physical implementations for information exchange. In the aforementioned literature, local information

flow has been widely used and well understood. However, because of the ignorance of the global objective, coordination between agents is often naive and undirected. With popularization of affordable wireless communication networks, it is possible to process and spread the global information in a distributed fashion using collective protocols. Thus, a uniform framework is needed to fully understand the formation control of networked multi-agent systems under the influence of local and global information flow.

### 1.2.2 Collective Behaviors and Consensus Seeking

Self-organizing features in animal groups, such as flocking, swarming, and schooling [57, 58], provide important insights for coordinated control in multi-agent systems. Theoretical discussions can be found in [59] for the self-driven particles alignment problem, in [60] where a continuum mechanics approach is used, and in [61] for rotating swarms with all-to-all interactions.

As we mentioned in the previous section, the traditional architecture for global information processing normally involves centralized information collection and computation. Surprisingly, collective algorithms provide possible distributed solutions for the same problem. A good example is average consensus seeking. Driving the states of all agents to a common value by distributed protocols based on a communication network is called the consensus problem. A popular discrete-time consensus protocol is proposed in [62] based on stochastic matrix theory [63]. Suppose  $x_i$  is the state of agent  $i$  and the protocol can be summarized as

$$x_i(k+1) = \sum_{j \in \mathcal{N}(i) \cup \{i\}} \alpha_{ij}(k) x_j(k) \quad (1.2)$$

where  $\mathcal{N}(i)$  represents the set of agents whose state is available to agent  $i$  at step  $k$ . We assume  $\alpha_{ij} \geq 0$  and  $\sum_j \alpha_{ij}(k) = 1$ . In other words, agent state is updated as the weighted average of its current value and its neighbors'. Correspondingly, a continuous-time consensus protocol can be found in [64, 65] as

$$\dot{x}_i(t) = - \sum_{j \in \mathcal{N}(i)} \beta_{ij}(t) (x_i(t) - x_j(t)) \quad (1.3)$$

where  $\beta_{ij}(t)$  denotes the positive weight. We say the system has achieved a consensus if  $\|x_i - x_j\|$  converges to zero as  $t \rightarrow \infty$  for any  $i \neq j$ . More specifically, it is called average consensus when  $x_i \rightarrow \sum x_i(0)/n$  as  $t \rightarrow \infty$  for any  $i$  where  $n$  is the number of agents.

Suppose the topology of the communication network is time-invariant. The necessary and sufficient condition for consensus protocol (1.2) and (1.3) to reach a consensus is that there exists a spanning tree in the topology [66, 67]. For average consensus seeking, the topology must be at least strongly connected and balanced [64]. When the topology is time-variant, such as in an ad-hoc wireless network, it has been shown that consensus protocols are still valid under switching topologies given the condition that at least one topology is strongly connected [65] or there exists a spanning tree [67] in each uniformly bounded time interval. The impact of communication delays on consensus seeking are also studied in [64, 68] where upper bounds of the delay margin are given for a fixed communication topology with uniform delay.

Consensus protocols has been employed in many engineering problems. For coordinated control, consensus schemes have been applied to achieve vehicle formations [66]. In rendezvous problems, consensus seeking is applied to control agents arrive at a certain location simultaneously [69]. Other applications include spacecraft attitude alignment [51], distributed decision making [70], asynchronous peer-to-peer networks [71], and robot synchronization [72]. All of these applications rely on the assumption that the convergence speed of consensus seeking is fast enough. A couple of methods have been reported to improve the convergence speed, such as finding the optimal weights associated with every communication link [37] or using random rewiring to change the topology [73], but they all face difficulties in practical implementation.

### 1.2.3 Control and Estimation over Networks

A unique feature of networked control systems (NCS) is that communication networks are employed to close the loops between system components such as sensors, controllers, and actuators. Figure 1.2 shows a typical diagram for NCS. A networked multi-agent system can be considered as a special case of NCS.

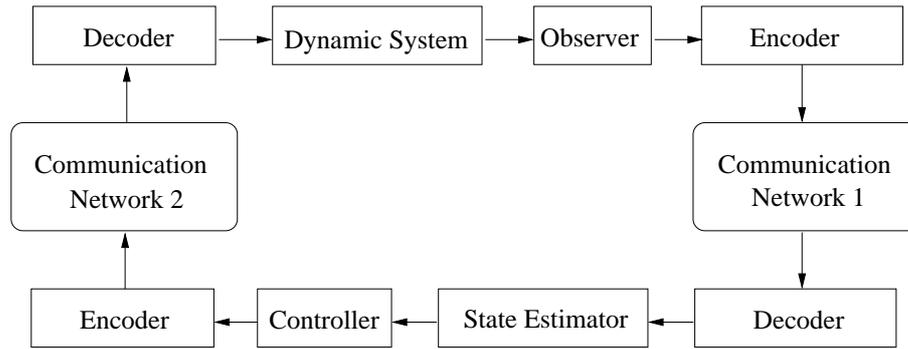


Figure 1.2: Diagram of a typical networked control system

In the structure of the NCS, a couple of fundamental assumptions for conventional control theory are not valid anymore due to the “imperfect” data links. For example:

- Infinite bandwidth. The communication channel can only transmit the data with certain precision under the constraint of limited bandwidth. Quantization and distortion must be considered for system design and analysis.
- Reliable connections. Sampled signals are transmitted in data packets that suffers from unreliability issues such as unpredictable transmission delays and random packet drops.
- Static structure. For networked multi-agent systems, dynamic routing and ad-hoc connectivity of modern communication networks makes the interaction topology time-variant and the coupling among agents may change as well.

Several of related research threads exist to accommodate these issues. One thread is designing communication protocols and algorithms that minimize the probability of network congestion and packet drops [74]. Another one is treating these issues as additional constraints. For example, the maximum allowable transfer interval is discussed in [75] for desired stability and performance. Methods to compensate for network-induced delays are presented in [76]. In [77], a “recursive state estimator” is used to generate minimum variance estimates in the presence of irregular communication delays.

During the last decade, there has been a surge of interest in combining tools from information and communication theory to solve “control problems with communication con-

straints.” Under the framework of NCS, many significant results are reported. The estimation and stabilization problem of a closed-loop dynamic system over a communication channel with finite bandwidth is first discussed in [78, 79]. The necessity of a unified approach to control, communication, and computation is emphasized in [80]. Quantization issues are discussed in [81]. In [82], an optimal algorithm quantizer, the “logarithmic quantizer”, is introduced in order to minimize the transmission bits. After that, a lower bound of the channel capacity needed to stabilize a linear time-invariant (LTI) system is given in [83]. Also, simple time-variant coding schemes are proposed for noiseless and noisy channels in [83] and [84], respectively. In [85], a concept of anytime capacity is proposed to deal with the real-time issue in NCS. Moreover, the authors for [86] try to address the stability of NCS in a stochastic way by introducing the concept of “almost-sure stability.”

For the model-based estimation problem, a popular method for linear systems is Kalman filtering [87, 88]. For a nonlinear system, an extended Kalman filter is a natural choice. In recent years, the moving horizon estimation approach [89] has become another promising method by transferring the estimation problem to a nonlinear optimization problem within a finite horizon time interval and solving it on-line. In NCS, the input data of estimator is disordered or intermittent because of the transmission delays or packet drops. Studies on filtering with intermittent observations can be tracked back to [90] and [91]. Other researchers try to model the Kalman filter with missing observations as jump linear systems (JLS), which are stochastic hybrid systems with linear dynamics and discrete Markov chains. Certain convergence criteria are given for expected estimation error covariance [92, 93].

More recently, the general case for Kalman filtering with intermittent observations has been studied in [94], where stochastic equations for error covariance are derived based on the i.i.d. Bernoulli packet-dropping model. A detailed analysis for the solutions of the modified algebraic Riccati equation is presented and statistical convergence properties are discussed. This work formally shows a phase transition phenomenon for the expected value of error covariance with respect to the packet-dropping rate. A similar approach is used in [95] to investigate state estimation problem with partial observation loss over sensor networks.

## 1.3 Statement of Contributions

Contributions of this work are briefly stated in following paragraphs, which are sorted by chapter and provide an outline of the thesis.

**Chapter 2** In this chapter, we review some basic concepts and ideas from algebraic graph theory. Preliminary results about the spectrum and infinity norm of the Laplacian matrix are also listed. Moreover, a brief description and common notations for quantization and distortion theory are given. Last, two stochastic models for random packet drops on packet-based communication networks are introduced in preparation for our discussion in Chapter 5.

**Chapter 3** In this chapter, we propose a novel control strategy for networked multi-agent formation control. In this scheme, every agent adjusts its behavior according to its neighbors as well as the global objective. Conditions on formation stability are discussed based on the connectivity of the interaction topology. This strategy is robust against the data link failure and greatly eases the design of local controllers. Moreover, the performance of disturbance resistance can be uniformly bounded independent of the size of the formation. Simulation and experimental results on the Caltech multi-vehicle wireless testbed are also given to verify the feasibility and advantages of this strategy.

**Chapter 4** In this chapter, we focus on the continuous model of consensus protocols in networked multi-agent systems. Based on the connectivity properties of the communication network, convergence conditions and final consensus values are presented. A new type of consensus protocol, the multi-hop relay protocol, is proposed. Without physically changing the topology of the network, this protocol generally increases the convergence speed. At the same time, time delays in communication networks are considered and a tradeoff between convergence speed and delay margin is discussed.

**Chapter 5** In this chapter, the problem of state estimation over a packet-based communication network is studied in which efficiency and reliability of the network are critical issues. The presence of packet drops and communication delays impair our ability to

measure and predict a dynamic process. Multiple description (MD) codes, a type of network source code, are used to compensate for packet drops. The benefits of MD codes include efficient bandwidth utilization and convergence of error covariance over a large set of packet-dropping rates, which are explicitly shown for two packet-dropping models: the i.i.d. model and the Markov chain model. Moreover, solutions of the generalized modified algebraic Riccati equation (MARE) are discussed and simulation results are presented.

**Chapter 6** In the last chapter, we summarize the results in this work and point out connections to other related areas. Future extensions and possible research directions are provided.

## Chapter 2

### Preliminaries

Basic concepts, notions, and mathematical tools that will be used through out this work are covered in this chapter. Several preliminary results are also listed. In order to be concise and consistent, we focus on those that are necessary for a clear understanding of following chapters. Readers who are familiar with graph theory and communication networks can skip this chapter safely.

Section 2.1 states basic ideas from algebraic graph theory that have been commonly used in computer science and communication networks, but may be unfamiliar to some potential readers. The spectrum of the Laplacian matrix and the infinity norm for the adjacency matrix are discussed as well. Results and properties in this section contribute to the results in Chapter 3 and 4.

Section 2.2 introduces concepts from rate distortion theory. General performance measurements for quantizers are introduced. The section provides necessary knowledge for readers to follow Chapter 5.

Section 2.3 gives two commonly used mathematical models for packet loss in modern digital communication networks. Both of them try to catch the stochastic characteristics for random packet drops. More details and model verifications can be found in [96, 97, 98].

## 2.1 Algebraic Graph Theory

### 2.1.1 Basic Concepts

A *directed graph*  $\mathcal{G} = (\mathcal{V}, \mathcal{E})$  is composed by a finite *vertex set*  $\mathcal{V}$  and an *edge set*  $\mathcal{E} \subseteq \mathcal{V}^2$ . Suppose there are  $n$  vertices in  $\mathcal{V}$ , each vertex is labelled by an integer  $i \in \{1, 2, \dots, n\}$  and  $n$  is the *order* of the graph. Each edge can be denoted by a pair of distinct vertices  $(v_i, v_j)$  where  $v_i$  is the *head* and  $v_j$  is the *tail*. If  $(v_i, v_j) \in \mathcal{E} \Leftrightarrow (v_j, v_i) \in \mathcal{E}$ , the graph is called *symmetric* or *undirected*. A graph  $\mathcal{G}$  is said to be *complete* if every possible edge exists.

For directed graph  $\mathcal{G}$ , the number of edges whose head is  $v_i$  is called the *out-degree* of node  $v_i$ . The number of edges whose tail is  $v_i$  is called the *in-degree* of node  $v_i$ . If edge  $(v_i, v_j) \in \mathcal{E}$ , then  $v_j$  is one of the *neighbors* of  $v_i$ . The set of neighbors of  $v_i$  is denoted by  $\mathcal{N}(i) = \{v_j \in \mathcal{V} : (v_i, v_j) \in \mathcal{E}\}$ .

A *strong path* in a directed graph is a sequence of distinct vertices  $[v_0, \dots, v_r]$  where  $(v_{i-1}, v_i) \in \mathcal{E}$  for any  $i \in \{1, \dots, r-1\}$  and  $r$  is called the *length*. A *weak path* is also a sequence of distinct vertices  $[v_0, \dots, v_r]$  as long as either  $(v_{i-1}, v_i)$  or  $(v_i, v_{i-1})$  belongs to  $\mathcal{E}$ .

Directed graphs over a vertex set can be categorized according to their connectivity properties. A directed graph is *weakly connected* if any two ordered vertices in the graph can be joined by a weak path, and is *strongly connected* if any two ordered vertices can be joined by a strong path. If a strongly connected directed graph is symmetric, then it is called *connected and symmetric*. If a directed graph is not weakly connected, then it is *disconnected*. Figure 2.1 reveals the relationship among these concepts.

A directed graph  $\mathcal{G}$  is called *acyclic* if it does not contain any edge cycles. In a directed acyclic graph, there exists at least one vertex that has zero out-degree. A *rooted directed spanning tree* for directed graph  $\mathcal{G}$  is a subgraph  $\mathcal{G}_r = (\mathcal{V}, \mathcal{E}_r)$  where  $\mathcal{E}_r$  is a subset of  $\mathcal{E}$  that connects, without any cycle, all vertices in  $\mathcal{G}$  so that each vertex, except the root, has one and only one outgoing edge. (Some literature defines this concept in the opposite direction, i.e., each vertex, except the root, has one and only one incoming edge.) Thus, a rooted directed spanning tree is acyclic. A *strong component* of directed graph  $\mathcal{G}$  is an induced subgraph that is maximal, subject to being strongly connected. Since a vertex is

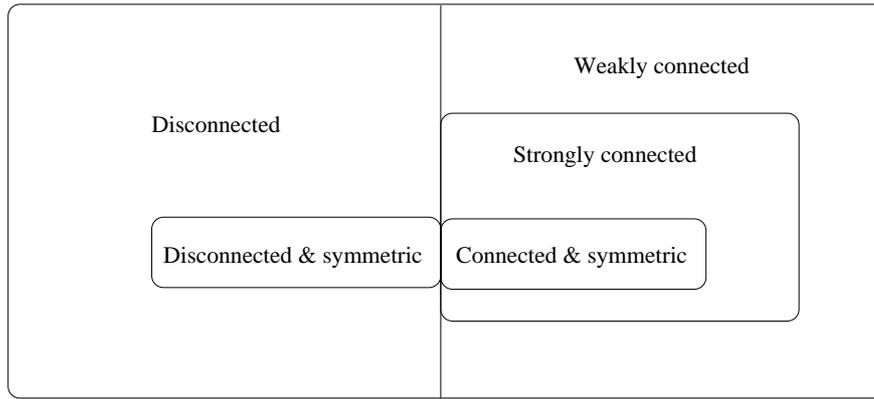


Figure 2.1: Classification for directed graphs

strongly connected, it follows that each vertex lies in a strong component. It can be shown that edges between any two strong components, if any, are uniformly directional, i.e., the heads of these edges belong to one component and tails belong to another. Otherwise, these two strong components compose a bigger strong component.

### 2.1.2 Matrices in Algebraic Graph Theory

Algebraic graph theory studies graphs in connection with linear algebra. Various matrices are naturally associated with the vertices and edges in a directed graph. Properties of graphs can be reflected in the algebraic analysis of these matrices. A couple of important matrices are introduced below.

An *adjacency matrix*  $\mathcal{A} = \{a_{ij}\}$  of a directed graph  $\mathcal{G}$  with order  $n$  is an  $n \times n$  matrix that is defined as

$$a_{ij} = \begin{cases} 1, & (v_i, v_j) \in \mathcal{E}; \\ 0, & \text{otherwise.} \end{cases} \quad (2.1)$$

Thus,  $\mathcal{A}$  is symmetric when  $\mathcal{G}$  is symmetric.

An *out-degree matrix*  $\mathcal{D} = \{d_{ij}\}$  of a directed graph  $\mathcal{G}$  with order  $n$  is an  $n \times n$  diagonal matrix as

$$d_{ii} = \sum_{j \neq i} a_{ij}. \quad (2.2)$$

It is clear that every diagonal element in  $\mathcal{D}$  equals the out-degree of the corresponding vertex. When  $\mathcal{G}$  is symmetric, the out-degree for each vertex equals the in-degree and  $\mathcal{D}$  is

also called the *degree matrix*.

Generally, a *weighted adjacency matrix*  $\mathcal{A} = \{a_{ij}\}$  is defined as

$$a_{ij} = \begin{cases} w_{ij}, & (v_i, v_j) \in \mathcal{E}; \\ 0, & \text{otherwise} \end{cases} \quad (2.3)$$

where  $w_{ij}$  is the positive weight associated with edge  $(v_i, v_j)$ . Then, the out-degree of node  $v_i$  is the sum of the weights of the edges whose head is  $v_i$ . The in-degree of node  $v_i$  is the sum of the weights of the edges whose tail is  $v_i$ . So the aforementioned adjacency matrix can be treated as a specific case of weighted adjacency matrix where all weights equal to 1.

A *Laplacian matrix*  $\mathcal{L}$  of a directed graph  $\mathcal{G}$  with order  $n$  is an  $n \times n$  matrix that is defined as

$$\mathcal{L} = \mathcal{D} - \mathcal{A}. \quad (2.4)$$

If we normalize each row of adjacency matrix by corresponding out-degree, we get the *normalized adjacency matrix* as

$$\bar{\mathcal{A}} = \mathcal{D}^{-1} \mathcal{A}. \quad (2.5)$$

In order to complete the definition, we set  $d_{ii}^{-1} = 0$  if a vertex  $v_i$  has zero out-degree. Moreover, we define the *normalized Laplacian matrix* as

$$\bar{\mathcal{L}} = \mathcal{D}^{-1} \mathcal{L}. \quad (2.6)$$

Other concepts include the *nonnegative* matrix if each element is nonnegative. Square matrix  $A$  is *reducible* if there exists a permutation matrix  $P$  such that  $PAP^T$  is block upper triangular as

$$PAP^T = \begin{bmatrix} A_{11} & 0 \\ A_{21} & A_{22} \end{bmatrix} \quad (2.7)$$

where  $A_{11}$  and  $A_{22}$  are square matrices.

A directed graph with seven vertices is shown in Figure 2.2 Part (a). It is clear that this directed graph is weakly connected. The associated adjacency matrix, out-degree matrix

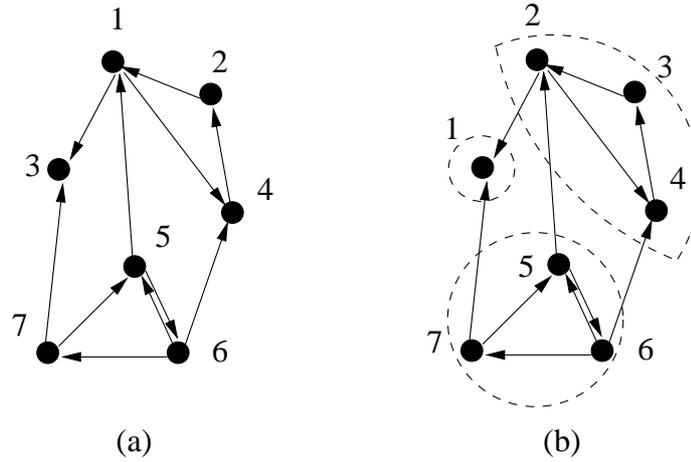


Figure 2.2: An example: directed graph with seven vertices

and Laplacian matrix are listed below:

$$\mathcal{A} = \begin{bmatrix} 0 & 0 & 1 & 1 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 & 1 & 0 & 1 \\ 0 & 0 & 1 & 0 & 1 & 0 & 0 \end{bmatrix} \quad \mathcal{D} = \begin{bmatrix} 2 & & & & & & \\ & 1 & & & & & \\ & & 0 & & & & \\ & & & 1 & & & \\ & & & & 2 & & \\ & & & & & 3 & \\ & & & & & & 2 \end{bmatrix}$$

$$\mathcal{L} = \begin{bmatrix} 2 & 0 & -1 & -1 & 0 & 0 & 0 \\ -1 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & -1 & 0 & 1 & 0 & 0 & 0 \\ -1 & 0 & 0 & 0 & 2 & -1 & 0 \\ 0 & 0 & 0 & -1 & -1 & 3 & -1 \\ 0 & 0 & -1 & 0 & -1 & 0 & 2 \end{bmatrix}.$$

The associated normalized adjacency matrix and normalized Laplacian matrix are

$$\bar{\mathcal{A}} = \begin{bmatrix} 0 & 0 & 1/2 & 1/2 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 1/2 & 0 & 0 & 0 & 0 & 1/2 & 0 \\ 0 & 0 & 0 & 1/3 & 1/3 & 0 & 1/3 \\ 0 & 0 & 1/2 & 0 & 1/2 & 0 & 0 \end{bmatrix}$$

and

$$\bar{\mathcal{L}} = \begin{bmatrix} 1 & 0 & -1/2 & -1/2 & 0 & 0 & 0 \\ -1 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & -1 & 0 & 1 & 0 & 0 & 0 \\ -1/2 & 0 & 0 & 0 & 1 & -1/2 & 0 \\ 0 & 0 & 0 & -1/3 & -1/3 & 1 & -1/3 \\ 0 & 0 & -1/2 & 0 & -1/2 & 0 & 1 \end{bmatrix}.$$

We change the vertex indices, as in part (b), and it is clear that there are three strongly components, composed by vertices sets  $\{1\}$ ,  $\{2, 3, 4\}$ , and  $\{5, 6, 7\}$ . The reducible adjacency matrix can be presented as

$$\mathcal{A} = \left[ \begin{array}{cccc|ccc} 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ \hline 1 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ \hline 0 & 1 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 & 1 & 0 & 1 \\ 1 & 0 & 0 & 0 & 1 & 0 & 0 \end{array} \right] = \begin{bmatrix} \mathcal{A}_{11} & & \\ \mathcal{A}_{21} & \mathcal{A}_{22} & \\ \mathcal{A}_{31} & \mathcal{A}_{32} & \mathcal{A}_{33} \end{bmatrix}$$

where  $\mathcal{A}_{11}$  is a scalar, and  $\mathcal{A}_{22}$  and  $\mathcal{A}_{33}$  are  $3 \times 3$  square matrices.

Next, we list some results about the rank and spectrum of the adjacency matrix and Laplacian matrix. Let  $\mathbf{1}_n$  and  $\mathbf{0}_n \in \mathcal{R}^n$  denote the vectors with all ones and all zeros, respectively. We start with strongly connected graphs.

**Corollary 2.1.1.** *Given a directed graph  $\mathcal{G}$  and its adjacency matrix  $\mathcal{A}$ ,  $\mathcal{G}$  is strongly connected if and only if  $\mathcal{A}$  is irreducible.*

This corollary is a direct result based on the properties of nonnegative matrices in [63].

**Theorem 2.1.1.** *Suppose the order of directed graph  $\mathcal{G}$  is  $n$  and the Laplacian matrix is  $\mathcal{L}$ . If  $\mathcal{G}$  is strongly connected, then  $\text{rank}(\mathcal{L}) = n - 1$ .*

The theorem states the connection between the connectivity of  $\mathcal{G}$  and the rank of  $\mathcal{L}$ . The converse of this theorem is not true. However, when  $\mathcal{G}$  is symmetric, the condition becomes necessary and sufficient. An extension is stated below:

**Corollary 2.1.2.** *A symmetric graph  $\mathcal{G}$  is connected if and only if  $\text{rank}(\mathcal{L}) = N - 1$ .*

Since the adjacency matrix is nonnegative, one fundamental theory for spectrum analysis is the Perron-Frobenius theorem [63]. It states that the spectral radius of a nonnegative and irreducible matrix is an algebraically simple eigenvalue, known as the Perron root. The eigenspace associated with the Perron root is one-dimensional. The unique positive eigenvector associated with the Perron root is called the Perron vector. The following properties that are collected from [99, 64, 66, 67, 100, 101] state features of the spectrum of adjacency matrix and Laplacian matrix.

**Property 2.1.2.** *Zero is an eigenvalue of  $\mathcal{L}$ , and  $\mathbf{1}_n$  is the associated right eigenvector.*

**Property 2.1.3.** *Given a directed graph  $\mathcal{G}$  and the associated Laplacian matrix  $\mathcal{L}$ :*

- *If  $\mathcal{G}$  is strongly connected, the zero eigenvalue of  $\mathcal{L}$  is simple;*
- *If  $\mathcal{G}$  is connected and symmetric,  $\mathcal{L}$  is symmetric and positive semi-definite. All eigenvalues are real and nonnegative, which can be written as*

$$0 = \lambda_1(\mathcal{L}) < \lambda_2(\mathcal{L}) \leq \dots \leq \lambda_n(\mathcal{L}).$$

**Property 2.1.4.** For a strongly connected graph  $\mathcal{G}$  with normalized adjacency matrix  $\bar{\mathcal{A}}$  and normalized Laplacian  $\bar{\mathcal{L}}$ ,

- $\lambda(\bar{\mathcal{L}}) = 1 - \lambda(\bar{\mathcal{A}})$ .
- $\lambda(\bar{\mathcal{L}})$  lies on a disk of radius 1 that is centered at the point  $1 + 0j$  in the complex plane.

For weakly connected graphs, we can reduce them by replacing each strong component with a vertex. Edges inside each component are discarded and edges between any two components are replaced by one single edge. So if  $\mathcal{G}$  is strongly connected, it can be reduced to a single vertex. If  $\mathcal{G}$  is weakly connected, it can be reduced to either a directed tree with a single root or a *forest* with multiple roots. If a strong component can be reduced to a root, then it is called a *root strong component*. For example, the directed graph in Figure 2.2 can be reduced to a directed tree, shown in Figure 2.3, and the root strong component has only one vertex.

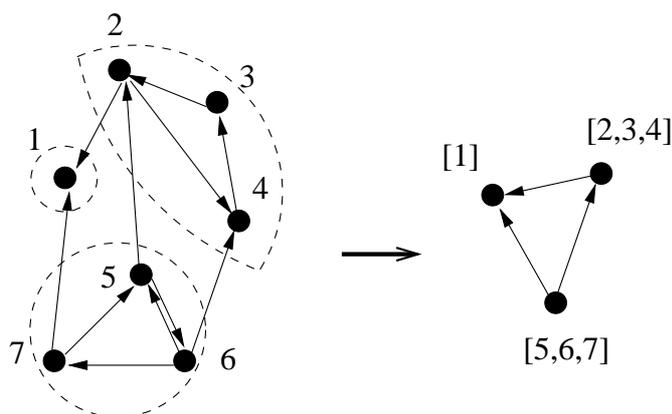


Figure 2.3: Reduction of a directed graph

According to the definition of  $\bar{\mathcal{D}}^{-1}$ ,  $\bar{\mathcal{L}} = I - \bar{\mathcal{A}}$  if and only if every vertex has nonzero out-degree. However, there exists at least one vertex with zero out-degree when  $\mathcal{G}$  is weakly connected. So we have the following result as the counterpart of Property 2.1.4 for weakly connected graphs.

**Lemma 2.1.1.** Suppose  $\mathcal{G}$  is weakly connected where vertices  $\{v_1, \dots, v_r\}$  have zero out-degree, then

- $\lambda(\bar{\mathcal{A}})$  still lies on the unit disk centered at the origin and  $\lambda(\bar{\mathcal{L}})$  is on a unit disk centered at the point  $1 + 0j$  in the complex plane.
- The first  $r$  eigenvalues of both of  $\bar{\mathcal{A}}$  and  $\bar{\mathcal{L}}$  are zeros.
- For other  $n - r$  eigenvalues,

$$\lambda(\bar{\mathcal{L}}) = 1 - \lambda(\bar{\mathcal{A}}).$$

**Proof** The first item is a directed extension based on the Geršgorin disk theorem. For others, we can rewrite  $\bar{\mathcal{A}}$  and  $\bar{\mathcal{L}}$  by separating these zero out-degree vertices from others. Suppose vertices  $\{v_1, \dots, v_r\}$  have zero out-degree, then

$$\bar{\mathcal{A}} = \begin{bmatrix} 0 & 0 \\ \bar{\mathcal{A}}_{21} & \bar{\mathcal{A}}_{22} \end{bmatrix}$$

where  $\bar{\mathcal{A}}_{22}$  is  $(n - r) \times (n - r)$ , and

$$\bar{\mathcal{L}} = \begin{bmatrix} 0 & 0 \\ -\bar{\mathcal{A}}_{21} & I - \bar{\mathcal{A}}_{22} \end{bmatrix}.$$

Thus, the eigenvalues of  $\bar{\mathcal{L}}$  are divided into two parts: the first  $r$  zeros and the left  $n - r$  from  $1 - \lambda(\bar{\mathcal{A}})$ . Then, the result follows.  $\square$

### 2.1.3 Infinity Norm of Normalized Adjacency Matrix

In this subsection, we discuss upper and lower bounds of the infinity norm of normalized adjacency matrix  $\bar{\mathcal{A}}$ . The infinity norm is defined by

$$\|\bar{\mathcal{A}}\|_{\infty} = \max_{x \neq 0} \frac{\|\bar{\mathcal{A}}x\|_{\infty}}{\|x\|_{\infty}} = \max_{1 \leq i \leq n} \sum_{j=1}^n |\bar{a}_{ij}|.$$

**Lemma 2.1.2.** *If directed graph  $\mathcal{G}$  is strongly connected, then*

$$\|\bar{\mathcal{A}}^m\|_{\infty} = 1 \tag{2.8}$$

for any positive integer  $m$ .

**Proof** Since  $\mathcal{G}$  is strongly connected, for every row in  $\bar{\mathcal{A}} = \{\bar{a}_{ij}\}$ ,  $\sum_j \bar{a}_{ij} = 1$ . When  $m = 2$ , the sum of the first row of  $\bar{\mathcal{A}}^2$  is

$$[\bar{a}_{11}, \bar{a}_{12}, \dots, \bar{a}_{1n}] \cdot \left( \begin{bmatrix} \bar{a}_{11} \\ \bar{a}_{21} \\ \vdots \\ \bar{a}_{n1} \end{bmatrix} + \begin{bmatrix} \bar{a}_{12} \\ \bar{a}_{22} \\ \vdots \\ \bar{a}_{n2} \end{bmatrix} + \dots + \begin{bmatrix} \bar{a}_{1n} \\ \bar{a}_{2n} \\ \vdots \\ \bar{a}_{nn} \end{bmatrix} \right) = [\bar{a}_{11}, \bar{a}_{12}, \dots, \bar{a}_{1n}] \cdot \mathbf{1}_n = 1.$$

This is true for any other row as well. With the induction to other  $m$ , the result follows.  $\square$

In most leader-follower cases, the leader's behavior normally is not affected by other followers, i.e., the out-degree of the leader is zero. Thus, the interaction topology is weakly connected and the "leader" is the unique root strong component of the graph. The following lemma gives out more precise bounds for  $\|\bar{\mathcal{A}}^m\|_\infty$  in that case:

**Lemma 2.1.3.** *Suppose  $\mathcal{G}$  is weakly connected with a single vertex as the unique root strong component; for any given  $\epsilon > 0$ , there exists a constant  $C$  such that*

$$\rho^m(\bar{\mathcal{A}}) \leq \|\bar{\mathcal{A}}^m\|_\infty \leq \begin{cases} 1, & m < n \\ C \cdot (\rho(\bar{\mathcal{A}}) + \epsilon)^{m-n+1}, & m \geq n \end{cases} \quad (2.9)$$

for any positive integer  $m$ , where  $n$  is the order of  $\mathcal{G}$  and  $\rho(\bar{\mathcal{A}})$  is the spectral radius of  $\bar{\mathcal{A}}$ .

**Proof** According to Geršgorin disk theorem [63], all eigenvalues of  $\bar{\mathcal{A}}$  are located in the unit disk, i.e.,

$$|\lambda_i| \leq 1, \forall i \in \{1, 2, \dots, n\}.$$

Then,

$$\rho(\bar{\mathcal{A}}) \leq 1 = \|\bar{\mathcal{A}}\|_\infty.$$

Since  $\rho(\bar{\mathcal{A}}^m) = \rho^m(\bar{\mathcal{A}})$ , we have

$$\rho^m(\bar{\mathcal{A}}) = \rho(\bar{\mathcal{A}}^m) \leq \|\bar{\mathcal{A}}^m\|_\infty.$$

Thus, the left half of inequality (2.9) is true.

For the right half, we have

$$\|\bar{\mathcal{A}}^m\|_\infty \leq \|\bar{\mathcal{A}}\|_\infty^m \leq 1$$

according to the definition of the norm. So one is an upper bound. What we need to do next is look for a tighter upper bound.

If the graph is acyclic, there exists an indexing method so that  $\bar{\mathcal{A}}$  is a lower triangular matrix and all diagonal elements are zeros, i.e.,  $\rho(\bar{\mathcal{A}}) = 0$ . Moreover,  $\bar{\mathcal{A}}^n = 0$  because the length of the largest strong path is  $n - 1$ . So we have

$$\|\bar{\mathcal{A}}^m\|_\infty = 0 = \rho^{m-n+1}(\bar{\mathcal{A}})$$

if  $m \geq n$ . In other words, the infinity norm jumps from 1 to 0 when  $m$  increases from  $n - 1$  to  $n$ .

If the graph is not acyclic, any loop in the graph can generate path with infinite length. In any loop, there exists at least one vertex whose out-degree is bigger than 1 because  $\mathcal{G}$  is weakly connected and the unique root strong component is a single vertex. Since every element in  $\bar{\mathcal{A}}^m$  is the gain of paths between certain pair of vertices with length  $m$ , all of the gains converge to zero as  $m \rightarrow \infty$ , so

$$\lim_{m \rightarrow \infty} \bar{\mathcal{A}}^m = 0. \quad (2.10)$$

According to [101], it is true that equation (2.10) holds if and only if all eigenvalues of  $\bar{\mathcal{A}}$  are inside the unit circle in the complex plane. Thus,  $\rho(\bar{\mathcal{A}}) < 1$ .

Since matrix  $\hat{A} = (\rho(\bar{\mathcal{A}}) + \epsilon)^{-1} \bar{\mathcal{A}}$  has spectral radius strictly less than 1,  $\hat{A}^m \rightarrow 0$  as  $m \rightarrow \infty$  and the elements of sequence  $\{\hat{A}^m\}$  should be bounded by a certain constant  $K$ . Then for  $m \geq n$ , we get

$$\begin{aligned} \|\bar{\mathcal{A}}^m\|_\infty &= \max_i \sum_{j=1}^n |(\bar{\mathcal{A}}^m)_{ij}| \\ &\leq nK(\rho(\bar{\mathcal{A}}) + \epsilon)^m \\ &\leq C(\rho(\bar{\mathcal{A}}) + \epsilon)^{m-n+1} \end{aligned}$$

and complete the proof. □

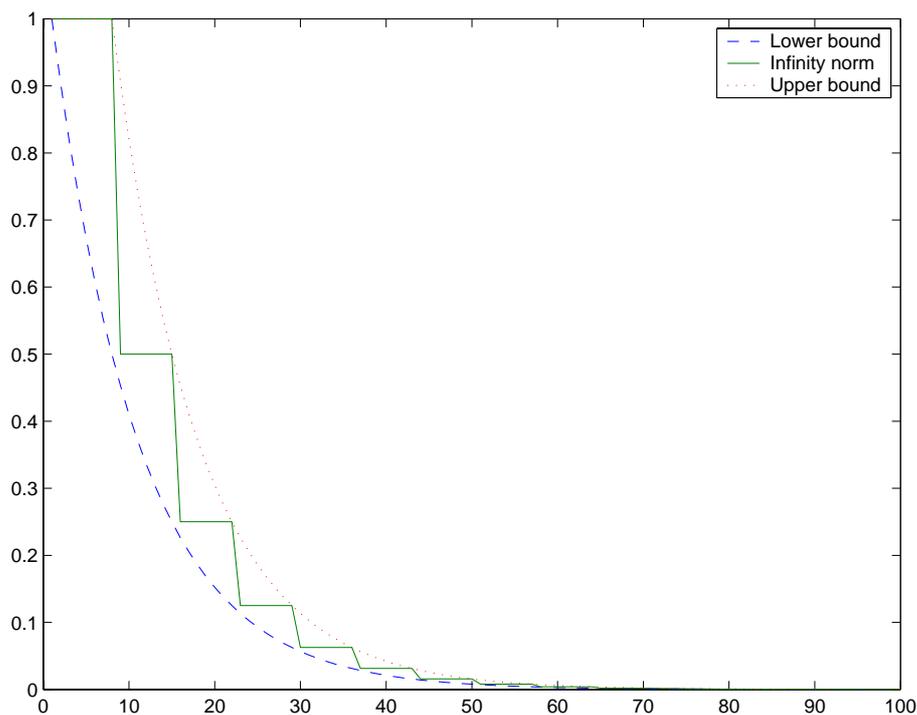


Figure 2.4: Bounds of normalized adjacency matrix norm

Figure 2.4 shows an example of the bounds of  $\bar{\mathcal{A}}$  of a leader-follower formation with weakly connected topology. With properly chosen parameters, the upper bound given in Lemma 2.1.3 is much tighter than the upper bound 1.

## 2.2 Quantization and Distortion

A simple model of *quantizer* usually consists of two parts: encoder and decoder. The encoder has a set of partitions  $\{V_1, V_2, \dots, V_M\}$ , which are called *quantization cells*, over the state space  $\mathcal{R}^m$ . The number of cells  $M$  is countable. Also,  $V_i \cap V_j = \emptyset$  for any  $i, j \in \{1, 2, \dots, M\}$  with  $i \neq j$ , and  $\bigcup_{i=1}^M V_i = \mathcal{R}^m$ . For any  $x \in \mathcal{R}^m$  as the input, the output of the encoder is an integer called the *quantization index*, which represents the quantization cell which  $x$  belongs to. So the encoder can be presented by a function as

$$q(x) = i \text{ if } x \in V_i. \quad (2.11)$$

The decoder has a *codebook* which defines a value  $c_i \in \mathcal{R}^m$  for every quantization index  $i$ , i.e.,

$$r(i) = c_i. \quad (2.12)$$

The *quantization rule* of a quantizer is the composite function

$$r \circ q(x) = c_i = \hat{x} \text{ if } x \in V_i \quad (2.13)$$

where  $\hat{x}$  is the *quantized representation* of  $x$ .

Apparently, quantization is a lossy process since part of the data resolution will be lost. For example, for any  $x_1, x_2 \in V_i$  and  $x_1 \neq x_2$ , we have  $r \circ q(x_1) = r \circ q(x_2) = c_i$ . Thus, a *distortion function*  $d(x, \hat{x}) : \mathcal{R}^m \times \mathcal{R}^m \rightarrow \mathcal{R}^+$  is defined as a measure of the cost of representing  $x$  by  $\hat{x}$ . One popular distortion function used for continuous state space is the *squared error distortion* defined on a value-by-value basis as

$$d(x, \hat{x}) = (x - \hat{x})^2. \quad (2.14)$$

If  $x$  is a random variable with the probability density function  $f(\cdot)$  over  $\mathcal{R}^m$ , the average distortion is given by

$$E[d(x, \hat{x})] = \int_{\mathcal{R}^m} d(x, \hat{x})f(x)dx = \sum_i \int_{V_i} d(x, \hat{x})f(x)dx. \quad (2.15)$$

The bits per source sample (bpss) of a quantizer is the number of average binary bits used to transmit a quantization index from encoder to decoder. When a fixed-length coding scheme is used, the bpss equals to  $R = \lceil \log_2 M \rceil$ . When a variable-length coding scheme is employed, the probability of each index  $i$  is  $p_i = \int_{V_i} f(x)dx$  and the average bpss is given by the entropy as

$$R = - \sum_{i=1}^M p_i \log_2 p_i. \quad (2.16)$$

A typical example of a variable-length code is the Huffman code [102].

The performance of a quantizer is evaluated by its average bpss and its distortion. Op-

timizing a quantizer involves minimizing both of these. For a fixed-length coding scheme, optimizing performance reduces to minimizing quantizer distortion and two conditions must be satisfied [103]:

- The nearest neighbor condition. A source sample must be mapped to the closest  $c_i$ ;
- The centroid condition. The value of  $c_i$  should be the centroid of cell  $V_i$ , i.e., the expected value of the source sample given that it lies in the cell.

For a variable-rate coding scheme, the centroid condition is still necessary for the quantizer to be optimal.

Moreover, a quantizer is called a vector quantizer when  $m > 1$  and is called a scalar quantizer when  $m = 1$ . The most popular quantizer is the *uniform scalar quantizer* where all cells are of equal size. The quantization rule of such a quantizer is characterized by three parameters: the size of its cells, which is often called the *step size* and denoted by  $\Delta$ , the number of the cells, and an *offset* between zero and  $\Delta$  that is used to locate partitions. The average distortion for uniform scalar quantizer with small step size is approximately  $\Delta^2/12$ .

## 2.3 Modeling for Packet Drops

For modern communication networks with high bit rates, packet drops can be caused by signal degradation over the medium (signal to noise ratio is too small), over-saturated network links (congestion in the network), faulty networking hardware, etc. A rich literature about packet drops can be found and many mathematic models have been proposed. In this section, two popular models that are mathematically simple yet sophisticated enough to capture the characteristics of packet drops in large scale networks are presented.

**Bernoulli Packet-Dropping Model** - As shown in Figure 2.5 part(a), the network is treated as a switch. A sequence of Bernoulli random variable  $\gamma_k$  is used to indicate whether the switch is on or off at time step  $k$ , i.e., packet  $k$  is transmitted through the communication network successfully or not. If it is on,  $\gamma_k = 1$ , otherwise,  $\gamma_k = 0$ . Those random variables

are independent and identically distributed (i.i.d.). This model is characterized by a single parameter  $\lambda$  that has the probability of  $\gamma_k$  being 1. So the model can be presented by

$$\gamma_k = \begin{cases} 1 & \text{with probability } \lambda \\ 0 & \text{with probability } 1 - \lambda \end{cases} \quad (2.17)$$

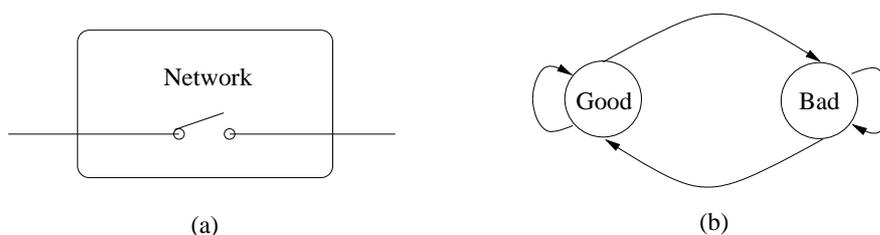


Figure 2.5: Models of packet drops: (a) i.i.d. Bernoulli model; (b) two-state Markov chain model

**Gilbert-Elliot Channel Model** - Sometimes packet drops occur in bursts due to the memory effect of the network. This bursty error behavior can be represented by a discrete Markov chain. The simplest one is the Gilbert-Elliot channel model [96, 97], or the two-state Markov chain model, which is shown in Figure 2.5, part (b). In this model, the network jumps between two possible states: “good” and “bad.” In the good state, the packet is transferred successfully; while in the bad state, the packet is dropped. The current network state,  $\mathcal{X}_k$ , depends only on the previous state  $\mathcal{X}_{k-1}$ . Assuming that 1 means the good state and 0 means the bad state, the transition probabilities are

$$\begin{cases} q_{01} = P[\mathcal{X}_k = 0 | \mathcal{X}_{k-1} = 1] \\ q_{11} = 1 - q_{01} \\ q_{10} = P[\mathcal{X}_k = 1 | \mathcal{X}_{k-1} = 0] \\ q_{00} = 1 - q_{10} \end{cases} \quad (2.18)$$

where  $q_{ij}$  is the transition probability from the previous state  $j$  to the current state  $i$ . Thus, the transition probability matrix  $\mathbf{Q}$  is given by

$$\mathbf{Q} = \begin{bmatrix} q_{00} & q_{01} \\ q_{10} & q_{11} \end{bmatrix}. \quad (2.19)$$

Clearly, the network has one-step “memory.” In order to capture a larger variety of temporal dependencies, a class of Markov processes with more states can be employed. However, for simplicity and without loss of generality, we use the two-state model in this work.

## Chapter 3

# Double-Graph Control Strategy for Formation Control

### 3.1 Formulation of Double-Graph Control Strategy

In this section, we formulate the problem of formation control for multi-agent systems and investigate the stability of the formation with respect to agent dynamics as well as the interaction topology.

#### 3.1.1 Gain Matrix in Signal Flow Graphs

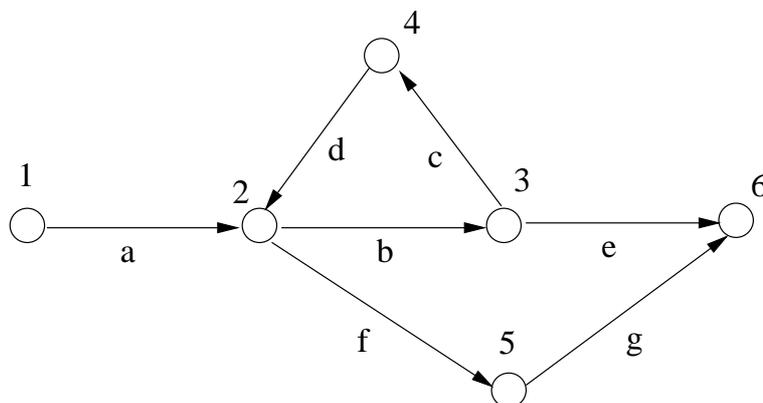


Figure 3.1: A signal flow graph

Suppose that each agent is a SISO system. One way to think about the interaction among agents is using linear signal flows [104]. A linear signal flow graph is treated as a weighted directed graph  $\mathcal{G}$ . The directions of edges indicate the directions of signals. The

weight associated with each edge represents the gain by which the signal is multiplied when passing the edge. Using Mason's gain formula, we can find out the actual gain from any input vertex to any output vertex. For multi-agent systems, we need to analyze all possible gains between any two vertices.

Let us start with a simple signal flow graph with six vertices as shown in Figure 3.1. Weights are listed along edges. By Mason's gain formula, we get the gain from vertex 1 to vertex 6 as

$$g_{16} = \frac{abe + afg}{1 - bcd}.$$

The following lemma introduces a square *gain matrix*  $G = \{g_{ij}\}$  where each element  $g_{ij}$  represents the gain from vertex  $v_i$  to vertex  $v_j$ .

**Lemma 3.1.1.** *For a signal flow graph  $\mathcal{G}$ , the gain matrix is*

$$G = (I - \mathcal{A})^{-1} \quad (3.1)$$

where  $\mathcal{A}$  is the weighted adjacency matrix associated with the signal flow graph.

**Proof** With a little abuse of notation, we use  $\mathcal{A}_{ij}$  to indicate the elements in  $\mathcal{A}$ . It is well known that elements in  $\mathcal{A}^m$  actually describe the sum of the gains of all paths with length  $m$  between any two vertices. For example,  $\mathcal{A}_{23} = b$  indicates there is only one path with length 1 between vertex 2 and 3 and the gain is  $b$ .  $\mathcal{A}_{26}^2 = be + fg$  indicates there exist two paths with length 2 between vertex 2 and 6 and the gains are  $be$  and  $fg$ .

If the signal flow graph doesn't include any loops, then  $\mathcal{A}^n = 0$  since the longest possible path has length  $n - 1$ . Otherwise,  $\mathcal{A}$  is not nilpotent. In Figure 3.1, the graph has a loop as  $2 \rightarrow 3 \rightarrow 4 \rightarrow 2$ . For the gain matrix  $G$ , it's true that  $G^6 \neq 0$ . For example,  $(1, 2, 3, 4, 2, 3, 4, 2, 3, 6)$  is a path from vertex 1 to 6 with length 9 and path gain  $ab^3c^2d^2e$ . Then, we write the gain matrix as

$$G = I + \mathcal{A} + \mathcal{A}^2 + \mathcal{A}^3 + \dots \quad (3.2)$$

by listing the gains of all possible paths. Since

$$\begin{aligned}
(I - \mathcal{A}) \cdot G &= (I - \mathcal{A}) \cdot (I + \mathcal{A} + \mathcal{A}^2 + \mathcal{A}^3 + \dots) \\
&= (I + \mathcal{A} + \mathcal{A}^2 + \mathcal{A}^3 + \dots) \\
&\quad - (\mathcal{A} + \mathcal{A}^2 + \mathcal{A}^3 + \dots) \\
&= I,
\end{aligned}$$

we have

$$Q = (I - \mathcal{A})^{-1}.$$

□

Thus, we easily get the weighted adjacency matrix of Figure 3.1 and the gain matrix as

$$\mathcal{A} = \begin{bmatrix} 0 & a & 0 & 0 & 0 & 0 \\ 0 & 0 & b & 0 & f & 0 \\ 0 & 0 & 0 & c & 0 & 0 \\ 0 & d & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & g \\ 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \quad G = \begin{bmatrix} 1 & \frac{a}{1-bcd} & \frac{ab}{1-bcd} & \frac{abc}{1-bcd} & \frac{af}{1-bcd} & \frac{abe+afg}{1-bcd} \\ 0 & \frac{1}{1-bcd} & \frac{b}{1-bcd} & \frac{bc}{1-bcd} & \frac{f}{1-bcd} & \frac{be+fg}{1-bcd} \\ 0 & \frac{cd}{1-bcd} & \frac{1}{1-bcd} & \frac{c}{1-bcd} & \frac{cdf}{1-bcd} & \frac{e+fgdc}{1-bcd} \\ 0 & \frac{d}{1-bcd} & \frac{bd}{1-bcd} & \frac{1}{1-bcd} & \frac{df}{1-bcd} & \frac{bde+fdg}{1-bcd} \\ 0 & 0 & 0 & 0 & 1 & g \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}$$

by the lemma. When we consider a multi-agent system, we treat it as a MIMO system and it is natural to analyze the gain matrix, i.e., the transfer function matrix. Suppose the input signal at vertex  $v_i$  is  $u_i$ . The outputs for the graph can be represented by

$$\begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_n \end{bmatrix} = G^T \cdot \begin{bmatrix} u_1 \\ u_2 \\ \vdots \\ u_n \end{bmatrix}. \quad (3.3)$$

This lemma requires that  $I - \mathcal{A}$  is not singular. Otherwise, at least one element in  $G$  is unbounded and the signal gain from  $v_i$  to  $v_j$  is infinite. For a multi-agent system, that means the formation is unstable.

### 3.1.2 Nyquist Criterion for Formation Stability

Suppose for a multi-agent system, there are  $n$  agents with identical linear dynamics denoted by

$$\dot{x}_i = Ax_i + Bu_i,$$

where  $i$  is the agent index. Each agent can get observations as

$$\begin{aligned} y_i &= C_1 x_i \\ z_{ij} &= C_2(x_i - x_j - L_{ij}) \end{aligned}$$

where  $j \in \mathcal{N}(i)$  and  $L_{ij}$  is the static offset based on the formation configuration. Based on the signals  $y_i$  and  $z_{ij}$ , designing a local controller with identical structure for each agent is a common distributed control strategy. A directed graph  $\mathcal{G}$  is used to represent the interaction topology. Each vertex represents one agent. If agent  $i$  can get the state of agent  $j$ , then there exists an edge  $(v_i, v_j)$  in  $\mathcal{G}$ . The directions of the edges are important. Usually, the errors for relative state measurements are synthesized into a single signal  $z_i$  with equal weights as

$$z_i = \frac{1}{d_{ii}} \sum_{j \in \mathcal{N}(i)} z_{ij}$$

where  $d_{ii}$  is the out-degree of vertex  $v_i$ .

According to [99], a local controller stabilizes the whole formation if and only if it simultaneously stabilizes the following  $n$  subsystems:

$$\begin{aligned} \dot{x} &= Ax + Bu \\ y &= C_1 x \\ z &= \lambda_i(\bar{\mathcal{L}})C_2 x \end{aligned}$$

where  $\lambda_i(\bar{\mathcal{L}})$  are the eigenvalues of the normalized Laplacian  $\bar{\mathcal{L}}$ .

However, it is difficult to get clear insights from these decoupled heterogeneous subsystems in the time domain when eigenvalues are complex. Another explanation for the stability criterion in the frequency domain are given below. Suppose each individual agent is a SISO system with local loop composed of a local controller  $C(s)$  and a plant model

$P(s)$ . The transfer function for a single agent is

$$H(s) = \frac{P(s)C(s)}{1 + P(s)C(s)}. \quad (3.4)$$

We can treat a multi-agent system as a signal flow graph with the inputs and outputs of each agent as signals. Figure 3.2 shows the interaction topology of a leader-follower multi-agent system where vertex 1 is the leader. The associated single flow graph is on the right side where  $y_i$  is the output of agent  $i$ . In the signal flow graph, the weight associated with edge  $(v_i, v_j)$  is  $H(s)/d_{jj}$ . Please note that directions of edges are changed in order to represent the directions of signals. It is easy to tell that the normalized adjacency matrices of these two graph are transposed, i.e.,  $\bar{\mathcal{A}}_{interaction} = \bar{\mathcal{A}} = \bar{\mathcal{A}}_{signal}^T$ .

Assuming that the input of agent 1 is signal  $u_1(s)$ . By Lemma 3.1.1, we get

$$\begin{aligned} \begin{bmatrix} y_1(s) \\ y_2(s) \\ \vdots \\ y_n(s) \end{bmatrix} &= (I - H(s) \cdot \bar{\mathcal{A}}_{signal}^T)^{-1} \cdot H(s) \begin{bmatrix} u_1(s) \\ 0 \\ \vdots \\ 0 \end{bmatrix} \\ &= (I - H(s) \cdot \bar{\mathcal{A}})^{-1} \cdot H(s) \begin{bmatrix} u_1(s) \\ 0 \\ \vdots \\ 0 \end{bmatrix}. \end{aligned} \quad (3.5)$$

In the general case, we have

$$\begin{bmatrix} y_1(s) \\ y_2(s) \\ \vdots \\ y_n(s) \end{bmatrix} = (I - H(s) \cdot \bar{\mathcal{A}})^{-1} \cdot H(s) \begin{bmatrix} u_1(s) \\ u_2(s) \\ \vdots \\ u_n(s) \end{bmatrix}. \quad (3.6)$$

Obviously, the multi-agent system is stable if and only if all of the transfer functions in Equation 3.6 are stable and  $H(s)$  is stable.

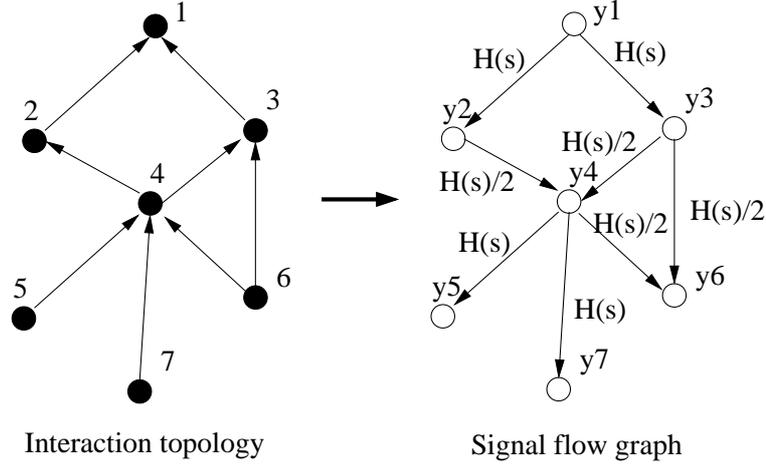


Figure 3.2: Interaction topology and the corresponding signal flow graph

**Lemma 3.1.2.** *The multi-agent system (3.6) is stable if and only if the net encirclement of the critical points  $-(1 - \lambda_i(\bar{\mathcal{A}}))^{-1}$  by the Nyquist plot of  $P(s)C(s)$  is zero.*

**Proof** We need the common denominator polynomial of the transfer function matrix to analyze the stability. According to the Schur decomposition theorem, there exists a unitary matrix  $T$  such that  $U = T^{-1}\bar{\mathcal{A}}T$  is upper triangular with the eigenvalues of  $\bar{\mathcal{A}}$  along the diagonal. The determinant of  $I - H(s) \cdot \bar{\mathcal{A}}$  is

$$\begin{aligned} \det(I - H(s) \cdot \bar{\mathcal{A}}) &= \det(T \cdot (I - H(s)U) \cdot T^{-1}) \\ &= \det(I - H(s)U) \\ &= \prod_{i=1}^n (1 - \lambda_i(\bar{\mathcal{A}})H(s)). \end{aligned}$$

Remember that the local closed-loop transfer function is

$$H(s) = \frac{P(s)C(s)}{1 + P(s)C(s)},$$

then

$$\det(I - H(s) \cdot \bar{\mathcal{A}}) = \prod_{i=1}^n \frac{1 + (1 - \lambda_i(\bar{\mathcal{A}}))P(s)C(s)}{1 + P(s)C(s)}.$$

So the common denominator polynomial is

$$\prod_{i=1}^n (1 + (1 - \lambda_i(\bar{\mathcal{A}}))P(s)C(s)). \quad (3.7)$$

According to the Nyquist criterion, the critical point  $-1$  is changed to  $n$  critical points  $-(1 - \lambda_i(\bar{\mathcal{A}}))^{-1}$ .  $\square$

When  $\mathcal{G}$  is strongly connected,  $1 - \lambda(\bar{\mathcal{A}})_i$  are the eigenvalues of  $\bar{\mathcal{L}}$  and the lemma is consistent with the results in [99]. Distribution patterns of the critical points can be summarized as:

- Suppose  $z$  is a critical point, then  $\text{Re}(z) \leq -1/2$ . In other words, all the critical points are located on the left side of the axis of  $-1/2$ . This is a directed extension from Property 2.1.4.
- Since  $1$  is an eigenvalue of  $\bar{\mathcal{A}}$ , then one of the critical points is at  $-\infty$ .
- If  $\mathcal{G}$  is symmetric, all critical points are real and no larger than  $-1/2$ .
- When  $\mathcal{G}$  is complete, all critical points except  $-\infty$  are located at  $-(n-1)/n$  coincidentally.

Intuitively, loops in  $\mathcal{G}$  introduce periodic forces among agents and jeopardize the formation stability. The bigger these loops are, the more separated these critical points are, and the harder the design of the local controller is.

When  $\mathcal{G}$  is weakly connected, the distribution patterns for the critical points are:

- For any critical point  $z$ , it is still true that  $\text{Re}(z) \leq -1/2$ .
- According to Lemma 2.1.3, at least one eigenvalue of  $\bar{\mathcal{A}}$  is zero. So  $-1$  is one of the critical points. Thus, stabilizing a single agent becomes a necessary condition for stabilizing the formation.
- If  $\mathcal{G}$  is acyclic, all critical points locate at  $-1$  coincidentally since all eigenvalues of  $\bar{\mathcal{A}}$  are  $0$ . In that case, stabilizing a single agent is the necessary and sufficient condition for stabilizing the formation.

### 3.1.3 Double-Graph Control Strategy

Most previous strategies for multi-agent formation control rely on direct agent interactions. Each agent adjusts its behavior according to its neighbors. This distributed approach

is inspired by on-board sensing technology and eases the collision avoidance issue because, most likely, neighbors in interaction topology are consistent with those agents within short geometric distance.

One difficulty of this strategy is that the formation stability is sensitive to the interaction topology when agent dynamics is nontrivial. According to Lemma 3.1.2, when the interaction topology changes, local controllers may destabilize the formation. A good example is given in [46] where an additional interaction link will destabilize the whole formation. In other words, we need to know the global knowledge of the interaction topology to design stabilizing local controllers. Another shortage is the poor disturbance resistance performance with respect to the size of the formation. Disturbance signals introduced at one agent can be magnified when they propagate to other coupled agents. That makes it impracticable to maintain a larger size formation. This effect is noticed in acyclic topologies at first and numerous works have been written on how to bound this accumulation, such as the “string stability” problem in vehicle platoons [22, 23, 105] and the “mesh stability” for “look-ahead” systems in [25]. However, a uniform approach to obtain good disturbance resistance for arbitrary interaction topology and general agent dynamics is unclear.

Due to incredible developments in communication technology, the ratio of the cost to the communication bandwidth has dropped dramatically in the recent couple of decades. It is possible to equip each agent with communication devices at low cost. Then a communication network is formed and information can be passed around over it, which gives more flexibility and possibility to coordinated control for multi-agent systems. Based on this thought, we propose the double-graph control strategy as follows.

Suppose a coordinated multi-agent system has  $n$  identical agents. As shown in Figure 3.3, a directed graph  $\mathcal{G}_1$  is employed to describe the *global coordination topology* that is used for global objective seeking and distribution. Suppose the global objective is  $O_{\text{global}}$  and the local copy of the global objective in agent  $i$  is  $O_i$ , then these local copies need to be synchronized over  $\mathcal{G}_1$ , i.e.,  $O_1 = O_2 = \dots = O_n = O_{\text{global}}$ . For formation control, static geometrical offsets need to be considered as well. Another directed graph  $\mathcal{G}_2$  is used to describe the *local interaction topology* where the dynamics of each agent is coupled with its neighbors.

The global coordination topology,  $\mathcal{G}_1$ , actually describes the topology of the communication network over agents. Each agent can “talk” and “listen.” Every edge in  $\mathcal{G}_1$  denotes a communication link between two agents. When the global objective  $O_{\text{global}}$  is pre-established, the most direct way for synchronization is broadcasting  $O_{\text{global}}$  over  $\mathcal{G}_1$ . One typical situation is the leader-follower formation control where the reference of the leader is a good choice of  $O_{\text{global}}$ . On the other hand, if the global objective is not clear beforehand, distributed collective protocols are used over  $\mathcal{G}_1$  to achieve the synchronization without centralized data collection and processing. Consensus protocols in [59, 64, 66], for instance, can be used for seeking  $O_{\text{global}}$  as a “virtual” reference of the formation structure. Currently, general approaches for collective protocol design and analysis attract many researchers.

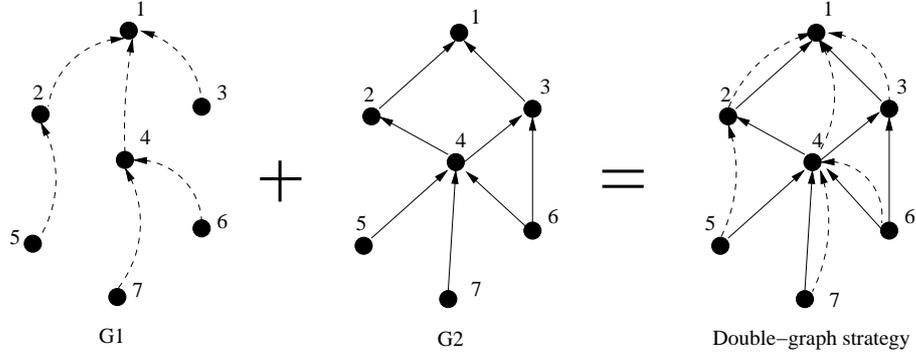


Figure 3.3: Double-graph control strategy for multi-agent systems

Based on the local interaction topology,  $\mathcal{G}_2$ , local controllers of the double-graph control strategy have identical structure shown in Figure 3.4. The dynamics for each agent is

$$Y_i(s) = H_1(s) \cdot O_i(s) + H_2(s) \cdot \sum_{j \in \mathcal{N}(i)} Y_{ij}(s) \quad (3.8)$$

where  $H_1(s)$  and  $H_2(s)$  are transfer functions,  $O_i(s)$  is the error signal based on the global objective, and  $Y_{ij}(s)$  is the outputs of agent  $i$ 's neighbors. In order to simplify the analysis, we choose transfer functions as

$$\begin{cases} H_1(s) = \alpha \cdot H(s) \\ H_2(s) = (1 - \alpha)/d_{ii} \cdot H(s) \end{cases} \quad (3.9)$$

where  $d_{ii}$  is the out-degree of vertex  $i$  in graph  $\mathcal{G}_2$  and  $0 \leq \alpha \leq 1$ . Thus,

$$Y_i(s) = H(s)(\alpha O_i(s) + \frac{1-\alpha}{d_{ii}} \sum_j Y_{ij}(s)). \quad (3.10)$$

That means the inputs of each agent based on  $\mathcal{G}_1$  and  $\mathcal{G}_2$  are weighted by coefficients  $\alpha$  and  $1 - \alpha$ , respectively.

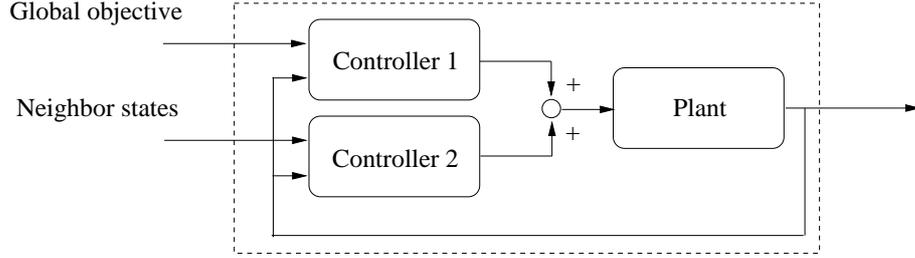


Figure 3.4: Diagram of the local control strategy

When  $d_{ii} = 0$ , i.e., there is no interaction between agent  $i$  and other agents, we choose

$$\begin{cases} H_1(s) = H(s) \\ H_2(s) = 0. \end{cases} \quad (3.11)$$

Then,

$$Y_i(s) = H(s)O_i(s). \quad (3.12)$$

Previous equations describe the structures of local controllers in double-graph control strategy. We will discuss the stability and performance in following sections.

## 3.2 Stability Analysis for Double-Graph Control Strategy

Let us start this section by a simple example. Suppose there are twelve agents in a leader-follower system. Vertex  $v_1$  is the leader and the others are followers. The agent dynamics is  $P(s) = 1/(s^2 + s + 4)$  and the local controller is  $C(s) = (800s + 2000)/(s + 40)$ . It can be shown that  $C(s)$  is a stabilizing controller for a single agent. However, the formation is not stable with the interaction topology in Figure 3.5 because there are two critical points

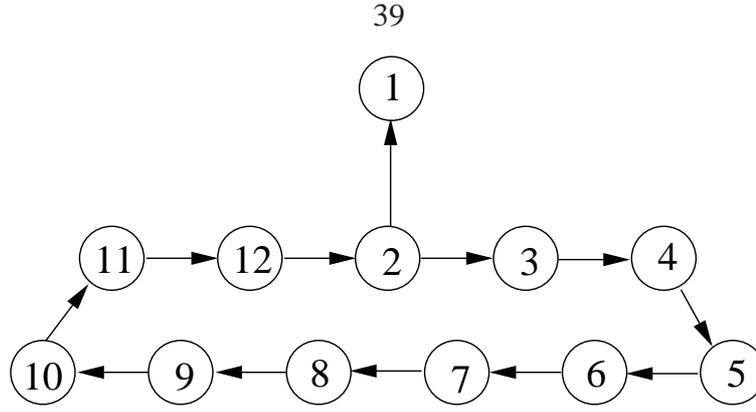


Figure 3.5: Interaction topology of a leader-follower system with twelve agents

encircled by the Nyquist plot of  $C(s)P(s)$  (shown in Figure 3.6).

According to our discussion in Section 3.1.2, the interaction topology determines the location of critical points. Thus, we may improve the formation stability using the following two methods:

- First, all the critical points are actually located in the left side of axis  $\text{Re}(x) = -1/2$ . If the Nyquist plot of  $P(s)C(s)$  stays in the right side of the axis  $\text{Re}(x) = -1/2$ , the formation is stable for any interactive topologies. However, that means that the infinity norm of transfer function  $H(s)$  is not bigger than 1 for any  $s = j\omega$ . Since the followers are tracking the leader with constant offsets, for general agent dynamics  $P(s)$ , this constraint is too strict to achieve due to the limitations of linear controller  $C(s)$  [105, 106].
- Second, a set of “stabilized” interaction topologies can be found for a certain local controller  $C(s)$  by limiting the eigenvalues of  $\bar{\mathcal{A}}$ . Actual interaction topology must belong to this graph set to make the formation stable. Of course, this method is not robust to link failures. Also, each vehicle must be aware of the global structure of the interaction topology.

If we employ the double-graph control strategy, we can avoid these problems. For double-graph control strategy, the formation stability includes two parts: the collective protocol stability over  $\mathcal{G}_1$  if  $O_{\text{global}}$  is not pre-established, and interaction stability over  $\mathcal{G}_2$ . In this chapter, we assume that synchronization of  $O_i$  is accomplished and focus on the interaction stability over  $\mathcal{G}_2$ .

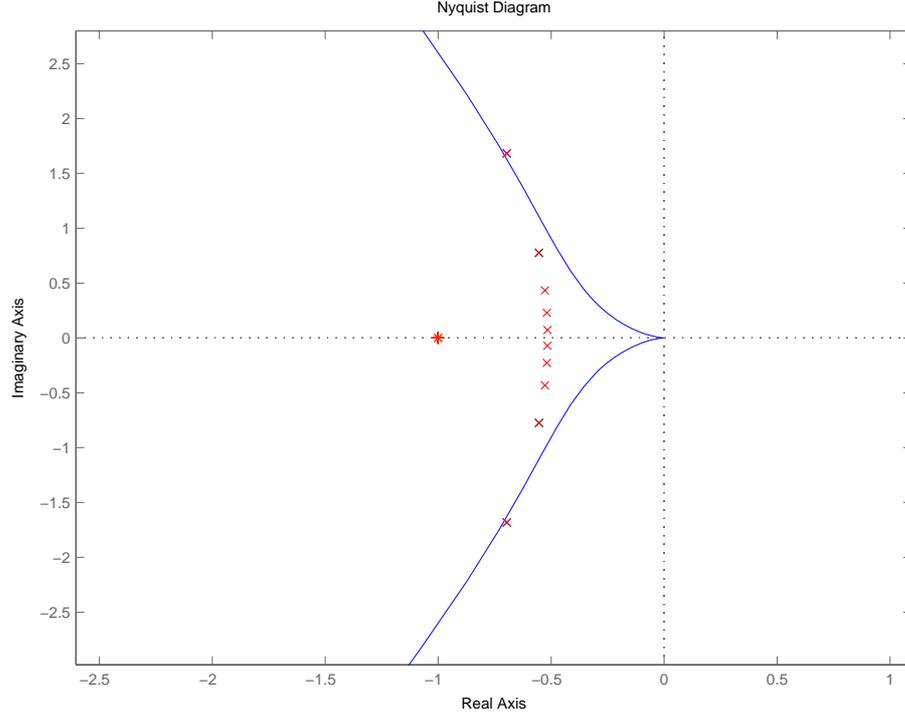


Figure 3.6: Nyquist plot and critical points

According to Equation (3.11) and (3.10), the transfer function matrix of a multi-agent system based on double-graph control strategy is

$$\begin{bmatrix} y_1(s) \\ \vdots \\ y_r(s) \\ y_{r+1}(s) \\ \vdots \\ y_n(s) \end{bmatrix} = (I - (1 - \alpha) \cdot H(s) \cdot \bar{\mathcal{A}}_2)^{-1} \cdot H(s) \cdot \begin{bmatrix} O_1(s) \\ \vdots \\ O_r s \\ \alpha \cdot O_{r+1}(s) \\ \vdots \\ \alpha \cdot O_n(s) \end{bmatrix} \quad (3.13)$$

where  $\bar{\mathcal{A}}_2$  is the normalized adjacency matrix of  $\mathcal{G}_2$  and the out-degrees of vertices  $v_1, \dots, v_r$  are zeros. Using the same reasoning as in Lemma 3.1.2, the common denominator polynomial for the transfer function matrix in Equation (3.13) is

$$\prod_{i=1}^n \left( 1 + (1 - (1 - \alpha)\lambda_i(\bar{\mathcal{A}}))P(s)C(s) \right). \quad (3.14)$$

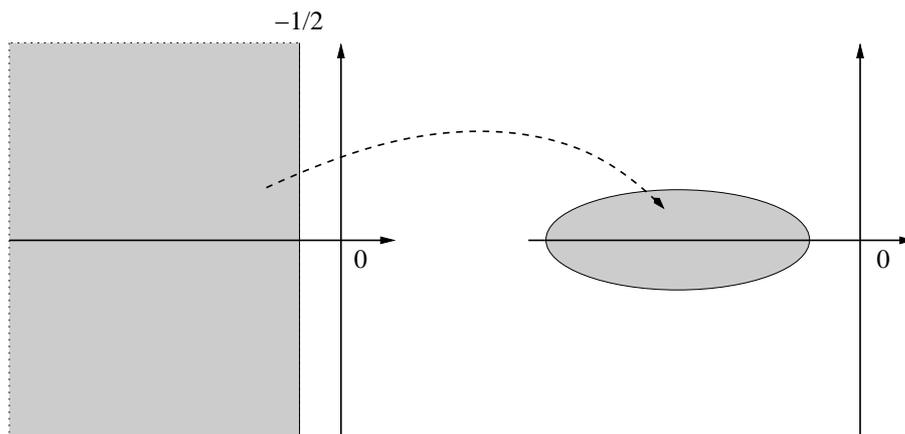


Figure 3.7: Mapping for critical points

Thus, the critical points are  $-1/(1 - \lambda_i(\bar{\mathcal{A}}_2) \cdot (1 - \alpha))$ . So the double-graph control strategy actually moves original critical points, which are  $-1/(1 - \lambda_i(\bar{\mathcal{A}}_2))$ , to new locations according to the mapping  $z \rightarrow z/(1 - \alpha z - \alpha)$ .

Actually, this mapping transfers the whole area on the left side of axis  $\text{Re}(x) = -1/2$  into a smaller ellipse. Figure 3.7 gives a geometric view of this mapping. The size of the ellipse can be adjusted by turning the value of  $\alpha$ . Figure 3.8 shows a couple of ellipses and the value of each corresponding  $\alpha$ . When  $\alpha$  changes from 0 to 1, the ellipse shrinks from the half complex plane to the single point  $-1 + 0 \cdot j$ . The bigger  $\alpha$  is, the smaller the ellipse is. This is reasonable since  $\alpha$  is bigger. Each agent more likely follows  $O_{\text{global}}$  and is less affected by its neighbors.

In the previous example, when the double-graph control strategy is used with  $\alpha = 0.5$ , the formation becomes stable with the same local controller. Figure 3.9 shows and the new locations of the critical points. It is clear that all critical points are located inside the ellipse and not encircled by the Nyquist plot.

**Theorem 3.2.1.** *For double-graph control strategy, the formation is stable if the local closed-loop transfer function  $H(s)$  satisfies the following conditions:*

- $H(s)$  is stable;
- $\|H(s)\|_{\infty} < \frac{1}{1-\alpha}$  if  $\mathcal{G}_2$  is not acyclic.

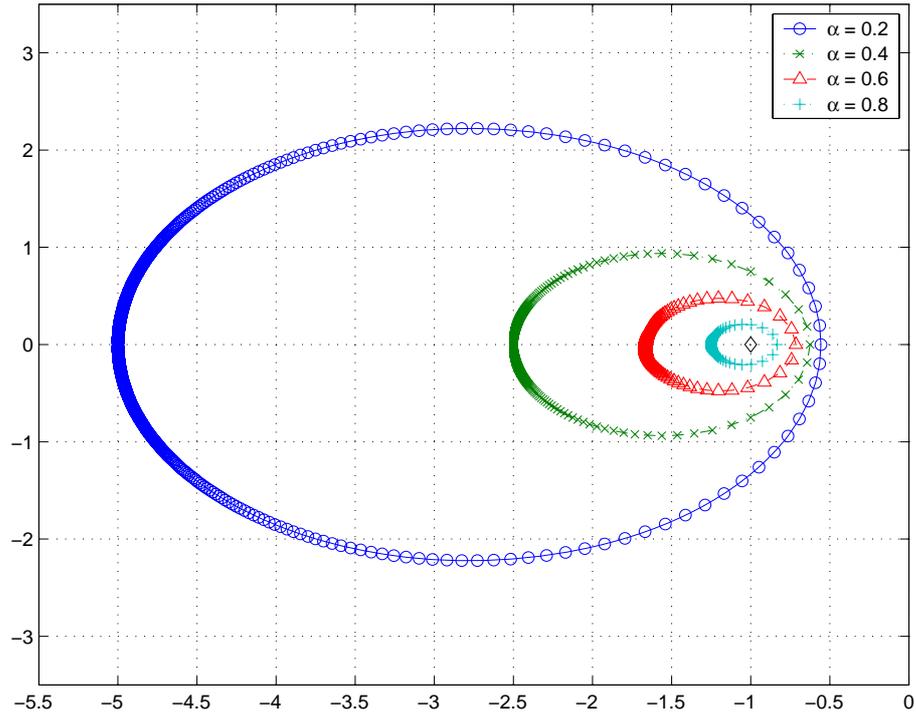


Figure 3.8: Ellipses for critical points with different  $\alpha$

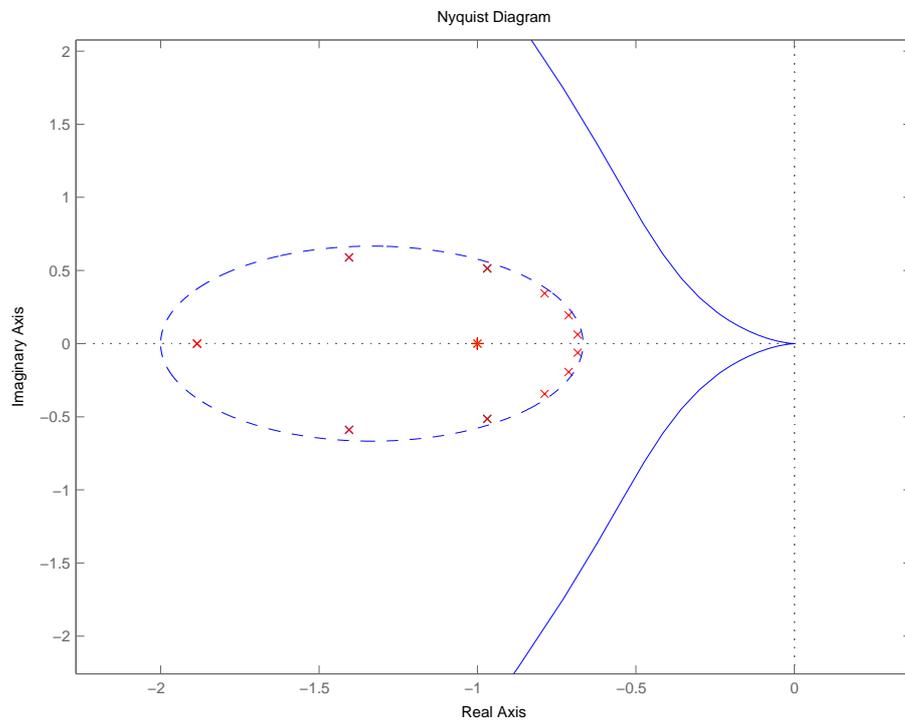


Figure 3.9: Nyquist plot and critical points for double-graph control strategy

**Proof** Since the formation dynamics is presented by Equation (3.13), then  $(I - (1 - \alpha) \cdot H(s) \cdot \bar{\mathcal{A}}_2)$  must be nonsingular for any  $s = j\omega$  and  $H(s)$  must be stable in order to make the system stable.

The determinant of  $(I - (1 - \alpha) \cdot H(s) \cdot \bar{\mathcal{A}}_2)$  is

$$\prod_{i=1}^n (1 - (1 - \alpha) \cdot \lambda_i(\bar{\mathcal{A}})H(s)).$$

When  $\mathcal{G}_2$  is acyclic,  $\lambda_i(\bar{\mathcal{A}}) = 0$  and the first condition is sufficient to stable the formation. When  $\mathcal{G}_2$  is not acyclic, we know that  $|\lambda_i(\bar{\mathcal{A}})| \leq 1$ . For any  $\lambda_i(\bar{\mathcal{A}}) \neq 0$ , if  $\|(1 - \alpha) \cdot H(s)\|_\infty < 1$ , then  $1 - (1 - \alpha) \cdot \lambda_i(\bar{\mathcal{A}})H(s) \neq 0$  for any  $\omega$ .  $\square$

This theorem gives the sufficient stabilizing conditions for double-graph control strategy. If  $\mathcal{G}_2$  is not acyclic, the conditions are conservative. The ellipses in Figure 3.8 are actually ‘‘M-curves’’ for the Nyquist plot [107] that is directly related to  $\|H(s)\|_\infty$ . When  $\|H(s)\|_\infty < 1/(1 - \alpha)$ , the Nyquist plot stays outside the ellipse. When  $\|H(s)\|_\infty = 1/(1 - \alpha)$ , the Nyquist plot coincides with the boundary of the ellipse at some points but never goes inside the ellipse. When  $\|H(s)\|_\infty > 1/(1 - \alpha)$ , the Nyquist plot cuts through the ellipse.

Moreover, this conservativeness can be treated as a robustness to the interaction topology  $\mathcal{G}_2$ . For double-graph control strategy, when  $\mathcal{G}_2$  changes, locations of the critical points change, but they are still inside the ellipse (refer to Lemma 2.1.3 and the mapping in Figure 3.7). So as long as the Nyquist plot stays outside, changes of  $\mathcal{G}_2$  do not affect the stability of the formation at all. This result is especially useful for coordinated formation control among mobile agents.

### 3.3 Performance for Double-Graph Control Strategy

In this section, we discuss the performance of disturbance resistance. Upper bounds of the disturbance output signals are given according to the connectivity of the interaction topology  $\mathcal{G}_2$ .

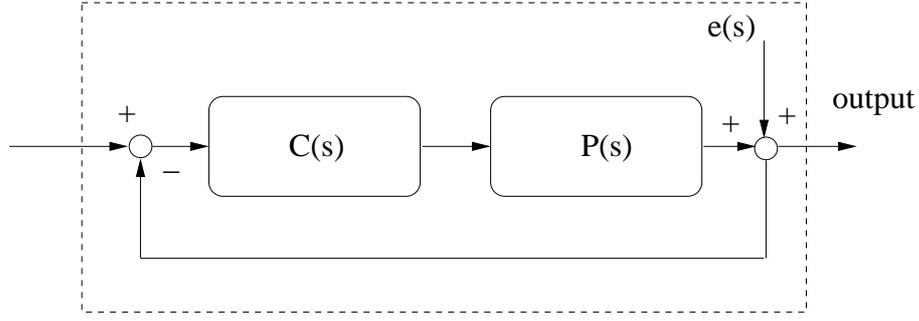


Figure 3.10: Disturbance for single agent

### 3.3.1 Disturbance Resistance with Acyclic Interaction Topologies

Suppose a disturbance signal  $e(s)$  is introduced at one agent  $i$  as shown in Figure 3.10. The output is  $\psi(s) = e(s)/(1 + P(s)C(s)) = e(s) \cdot S(s)$  where  $S(s)$  is the transfer function from  $e(s)$  to  $\psi(s)$ . In multi-agent systems, this diminished disturbance  $\psi(s)$  acts as inputs for other directly coupled agents and flows through the formation. When the interaction topology is weakly connected and acyclic, the following lemma gives an upper bound to quantify this disturbance signal over the whole formation.

**Lemma 3.3.1.** *For double-graph control strategy, assume  $\mathcal{G}_2$  is weakly connected and acyclic. The disturbance signal on any agent  $i$  is bounded by*

$$\|\psi_i(s)\|_\infty \leq \|S(s) \cdot e(s)\|_\infty$$

for any  $i \in \{1, \dots, n\}$  if  $\|H(s)\|_\infty < 1/(1 - \alpha)$ .

**Proof** Since  $\mathcal{G}_2$  is acyclic,  $\bar{\mathcal{A}}_2$  is nilpotent with  $\bar{\mathcal{A}}_2^n = 0$ . Also the matrix  $M = \bar{\mathcal{A}} + \bar{\mathcal{A}}^2 + \dots + \bar{\mathcal{A}}^{n-1}$  is lower triangular with zero diagonal elements if we label the vertices properly.

Next, we show that any nonzero element in  $M$  is no larger than 1. In order to make this point clear, we assign positive weight  $d_{jj}$  (the out-degree of  $v_j$ ) to edge  $(v_i, v_j)$  in  $\mathcal{G}_2$ . The value of  $M_{ij}$  is the sum of the gains of all possible paths from  $v_i$  to  $v_j$ . If there exists an element  $M_{ij} > 1$ , then for  $k \in \mathcal{N}(i)$ , there exists at least one  $M(kj) > 1$  since

$$M_{ij} = 1/d_{ii} \sum_{k \in \mathcal{N}(i)} M(kj).$$

We choose the biggest  $M_{kj}$  and repeat the searching. Because  $\mathcal{G}_2$  is acyclic, the searching process reaches the final stage after at most  $n - 1$  rounds and there exists at least a  $M_{lj} > 1$  where vertices  $v_l$  and  $v_j$  are adjacent. That is against the definition of  $\bar{\mathcal{A}}$ .

Now, we are ready to get the upper bound. According to Lemma 3.1.1, the transfer function matrix for the disturbance signal is

$$\begin{bmatrix} \psi_1(s) \\ \vdots \\ \psi_i(s) \\ \vdots \\ \psi_n(s) \end{bmatrix} = \left( I + (1 - \alpha)H(s)\bar{\mathcal{A}} + \cdots + ((1 - \alpha)H(s)\bar{\mathcal{A}})^{n-1} \right) \cdot \begin{bmatrix} \vdots \\ 0 \\ S(s) \cdot e(s) \\ 0 \\ \vdots \end{bmatrix}.$$

So for any agent  $j \neq i$ , the infinity norm of the disturbance is

$$\begin{aligned} \|\psi_j(s)\|_\infty &\leq \left( \|(1 - \alpha)H(s)\bar{\mathcal{A}}_{ji}\|_\infty \right. \\ &\quad + \|(1 - \alpha)^2 H(s)^2 \bar{\mathcal{A}}_{ji}^2\|_\infty + \cdots \\ &\quad \left. + \|(1 - \alpha)^{n-1} H(s)^{n-1} \bar{\mathcal{A}}_{ji}^{n-1}\|_\infty \right) \cdot \|S(s)e(s)\|_\infty \\ &\leq (\bar{\mathcal{A}}_{ji} + \cdots + \bar{\mathcal{A}}_{ji}^{n-1}) \cdot \|(1 - \alpha)H(s)\|_\infty \cdot \|S(s)e(s)\|_\infty \\ &= M_{ji} \cdot \|(1 - \alpha)H(s)\|_\infty \cdot \|S(s)e(s)\|_\infty \\ &\leq \|S(s)e(s)\|_\infty. \end{aligned}$$

For agent  $i$ , it is true that  $\|\psi_i(s)\|_\infty = \|S(s)e(s)\|_\infty$ .  $\square$

This lemma states that, by using double-graph control strategy, the infinity norm of the disturbance signal generated on any other agent is not amplified. This result is independent of the number of agents in the formation. Thus, error accumulation phenomena reported in [22, 25] are completely eliminated due to the relative “weak” coupling through the interaction topology.

### 3.3.2 Disturbance Resistance with Arbitrary Interaction Topologies

For arbitrary interaction topology, the problem is generalized to disturbance resistance in a MIMO system where any agent can introduce a disturbance signal. The problem is

formulated as

$$\Psi(s) = (I - (1 - \alpha)H(s) \cdot \bar{\mathcal{A}}_2)^{-1} \cdot S(s) \cdot E(s) \quad (3.15)$$

where

$$\Psi(s) = \begin{bmatrix} \psi_1(s) \\ \psi_2(s) \\ \vdots \\ \psi_n(s) \end{bmatrix} \quad \text{and} \quad E(s) = \begin{bmatrix} e_1(s) \\ e_2(s) \\ \vdots \\ e_n(s) \end{bmatrix}.$$

The infinity norm is used to measure signal vectors as

$$\begin{cases} \|\Psi(s)\|_\infty = \max_{i=1}^n \|\psi_i(s)\|_\infty \\ \|E(s)\|_\infty = \max_{i=1}^n \|e_i(s)\|_\infty. \end{cases}$$

**Theorem 3.3.1.** *For double-graph control strategy with stable local transfer function  $H(s)$ , suppose  $\|(1 - \alpha) \cdot H(s)\|_\infty = W < 1$ . Then*

$$\frac{\|\Psi(s)\|_\infty}{\|E(s)\|_\infty} \leq \frac{1}{1 - W} \cdot \|S(s)\|_\infty.$$

**Proof** According to Theorem 3.2.1,  $\|(1 - \alpha) \cdot H(s)\|_\infty < 1$  means the transfer function matrix in Equation (3.15) is stable. Rewrite  $\|(I - (1 - \alpha)H\bar{\mathcal{A}}_2)^{-1}\|_\infty$  as

$$\begin{aligned} & \|I + (1 - \alpha)H(s)\bar{\mathcal{A}}_2 + (1 - \alpha)^2 H(s)^2 \bar{\mathcal{A}}_2^2 + \dots\|_\infty \\ & \leq \|I\|_\infty + \|(1 - \alpha)H(s)\|_\infty \cdot \|\bar{\mathcal{A}}_2\|_\infty + \|(1 - \alpha)H(s)\|_\infty^2 \cdot \|\bar{\mathcal{A}}_2^2\|_\infty + \dots \\ & \leq 1 + W\|\bar{\mathcal{A}}_2\|_\infty + W^2\|\bar{\mathcal{A}}_2^2\|_\infty + \dots \end{aligned} \quad (3.16)$$

As we discuss in Lemma 2.1.1 and 2.1.2, the upper bound of  $\|\bar{\mathcal{A}}_2^m\|_\infty$  is 1 for any connected interaction topology. So the infinity norm of the transfer function matrix is upper bounded by  $1/(1 - W)$ .  $\square$

As we mentioned in Chapter 2, in the leader-follower case with a unique leader, the infinity norm of  $\bar{\mathcal{A}}_2^m$  converges to zero since  $\rho(\bar{\mathcal{A}}_2) < 1$ . Thus, the discussion for  $\|\bar{\mathcal{A}}_2^m\|_\infty$  given in Lemma 2.1.2 results in a tighter upper bound for the infinity norm of disturbance signals.

**Corollary 3.3.1.** *When the interaction topology  $\mathcal{G}_2$  is weakly connected with a single vertex as the unique root strong component, the upper bound in Theorem 3.3.1 changes to*

$$\frac{\|\Psi(s)\|_\infty}{\|E(s)\|_\infty} \leq \left( \frac{1 - W^n}{1 - W} + \frac{C \cdot (\rho(\bar{\mathcal{A}}_2) + \epsilon)W^n}{1 - (\rho(\bar{\mathcal{A}}_2) + \epsilon)W} \right) \cdot \|S(s)\|_\infty.$$

**Proof** According to Lemma 2.1.2, the sum of the first  $n$  term in Inequality (3.16) is no larger than

$$\frac{1 - W^n}{1 - W}.$$

The rest terms composes the left part in the upper bound

$$\frac{C \cdot (\rho(\bar{\mathcal{A}}_2) + \epsilon)W^n}{1 - (\rho(\bar{\mathcal{A}}_2) + \epsilon)W}.$$

Then, the upper bound follows. □

### 3.3.3 Other Performance Issues

Other than the disturbance resistance, two more performance issues related to the double-graph control strategy are listed below:

**Collision Avoidance:** In formation control, interactions between nearby agents provide “separation” and “cohesion” feedbacks that keep the formation in shape. When  $\alpha$  becomes bigger, these interactions become weaker. Thus, each agent is less capable of reacting to the changes in its local environment and formation maintenance relies more on the synchronization of the global objective.

In tight formations, these direct interactions between agents are very important for collision avoidance. When  $\alpha$  approaches to 0, the coupling between agents is “strong,” which generates quick responses to neighbors’ unwanted movements, but the disturbance gain  $(1 - \alpha)H(s)$  becomes larger. On the other hand, if  $\alpha$  goes to 1, each agent adjusts its behavior according to the local copy  $O_i$  and pays less attention to its neighbors. It is clear that there exists a tradeoff between the disturbance resistance and collision avoidance. Coefficient  $\alpha$  is an indicator of this tradeoff. Detailed quantitative analysis is out of the scope of this work.

**Time Delays:** We classify time delays in formation control into three types.

- The first type is the delay inside the local control loop. This delay can be treated as part of the model of the single agent. So it will show in the local transfer function  $H(s)$  and be handled by conventional methods.
- The second type is in the interaction topology  $\mathcal{G}_2$ . Each agent accesses its neighbors' states with delays. Each edge can be replaced by a transfer function as  $e^{-\tau_{ij}s}$ . Suppose all  $\tau_{ij}$  are identical and equal to  $\hat{\tau}$ , then the transfer function matrix changes to

$$(I - (1 - \alpha) \cdot H(s) \cdot e^{-\hat{\tau}s} \cdot \bar{\mathcal{A}}_2)^{-1} \quad (3.17)$$

and

$$\det(I - (1 - \alpha) \cdot H(s) \cdot e^{-\hat{\tau}s} \cdot \bar{\mathcal{A}}_2) = \prod_{i=1}^n (1 - (1 - \alpha) \cdot \lambda_i(\bar{\mathcal{A}}_2) H(s) \cdot e^{-\hat{\tau}s}).$$

When  $\|(1 - \alpha)H(s)\|_\infty < 1$ , we know that  $\|e^{-\hat{\tau}s}\|_\infty = 1$  and the determinant cannot be zero. Thus, the transfer function matrix is stable for any  $\hat{\tau}$ . So, uniform delays in  $\mathcal{G}_2$  do not affect the formation stability.

- The third type is the communication delays in the global coordination topology  $\mathcal{G}_1$ . Those delays only appear in the input term in Equation (3.13) and do not affect the stability either. However, the synchronization of  $O_i$  is difficult to achieve. We will discuss this issue in Chapter 4.

## 3.4 Simulation and Experimental Results

### 3.4.1 Multi-Vehicle Wireless Testbed

The Caltech Multi-Vehicle Wireless Testbed (MVWT) [108] is an experimental platform for validating theoretical advances in multi-agent coordinated and cooperative control. It consists of eight fan-driven vehicles that glide on low-friction omni-directional casters. Each vehicle is equipped with an on-board computer, local sensors, and a wireless card.

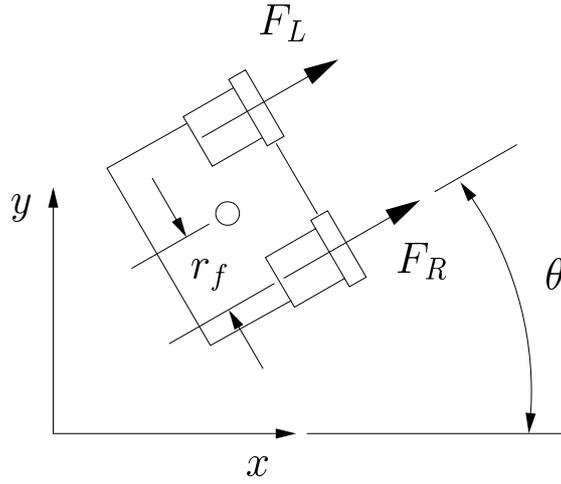


Figure 3.11: Schematic plot of MVWT vehicle.

Vehicles can communicate with each other over a local wireless network. A “hat” is put on the top of each vehicle so that the MVWT lab positioning system (LPS) can locate its position. A unique feature of this testbed is that the vehicles are under-actuated and exhibit nontrivial dynamics. This forces us to actively stabilize the vehicles while also trying to accomplish cooperative and coordinated tasks.

The simplified dynamics (assuming perfect sensing and actuation, no delays, no disturbances, and linear friction) for the vehicle are listed below. They are derived by observations from the simple schematic of the vehicle shown in Figure 3.11.

$$\begin{aligned}
 m\ddot{x} &= -\mu\dot{x} + (F_R + F_L)\cos\theta \\
 m\ddot{y} &= -\mu\dot{y} + (F_R + F_L)\sin\theta \\
 J\ddot{\theta} &= -\psi\dot{\theta} + (F_R - F_L)r_f.
 \end{aligned} \tag{3.18}$$

These equations include four physical parameters: the mass  $m$ , the mass moment of inertia  $J$ , and the linear and rotational viscous friction coefficients  $\mu$  and  $\psi$ . The geometric parameter  $r_f$  is the distance between the center of mass of the vehicle and each fan axis. The coordinate frame is inertial and the forces  $F_L$  and  $F_R$  are applied at the fan axes. We take  $\dot{x}_e, \dot{y}_e, \dot{\theta}_e$  as state variables and linearize the dynamics at the equilibrium point  $[C, 0, 0, 0, 0]$ . The linearized error dynamics is:

$$\begin{bmatrix} \ddot{x}_e \\ \dot{y}_e \\ \dot{y}_e \\ \dot{\theta}_e \\ \ddot{\theta}_e \end{bmatrix} = \begin{bmatrix} -\frac{\mu}{m} & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & -\frac{\mu}{m} & \frac{C\mu}{m} & 0 \\ 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & -\frac{\psi}{J} \end{bmatrix} \begin{bmatrix} \dot{x}_e \\ y_e \\ \dot{y}_e \\ \theta_e \\ \dot{\theta}_e \end{bmatrix} + \begin{bmatrix} \frac{1}{m} & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & \frac{r_f}{J} \end{bmatrix} \begin{bmatrix} u_1 \\ u_2 \end{bmatrix}$$

where  $u_1 = F_R + F_L - N$  and  $u_2 = F_R - F_L$  are control laws,  $N = C\mu$  is the feed-forward signal, which maintains the constant speed. We also get the transfer functions as

$$\frac{\dot{x}_e(s)}{u_1(s)} = \frac{1}{ms + \mu},$$

$$\frac{y_e(s)}{\theta(s)} = \frac{C\mu}{ms^2 + \mu s},$$

$$\frac{\theta_e(s)}{u_2(s)} = \frac{r_f}{Js^2 + \psi s}.$$

Please note that  $\dot{x}_e$  and  $u_1$  are decoupled from  $y_e$ ,  $\dot{\theta}_e$ , and  $u_2$ . Thus, we treat this system as two parts:

- Speed dynamics. The speed  $\dot{x}_e$  is controlled by  $u_1$ . This is a first-order dynamics.
- Lateral dynamics. The lateral  $y_e$  is effected by  $\theta_e$ , which is driven by  $u_2$ .

A PI controller,  $C_v = 8 + 5/s$ , is used to control the speed. Figure 3.12 shows the two-layer control diagram for the lateral dynamics. In order to make the inner layer response much faster than the outer layer, we design the controllers separately as

$$C_{in} = 60(s + 2)/(s + 25),$$

$$C_{out} = (40/C) \cdot (s + 0.5)/(s + 30).$$

### 3.4.2 Simulation Results

Simulation results based on the vehicle model are listed below. There are five vehicles in the formation. Vehicle 1 is the leader and the others are followers. Thus, the global

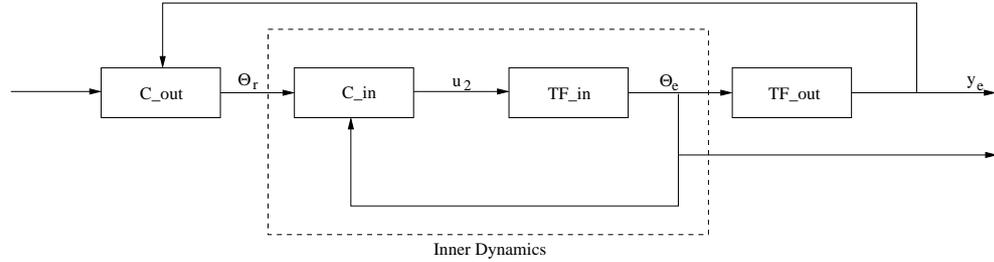


Figure 3.12: Control diagram for lateral dynamics.

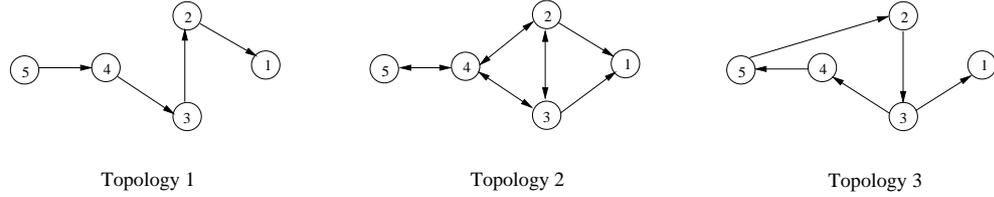


Figure 3.13: Three interaction topologies of vehicle formation

objective is the reference input of vehicle 1. For the interaction topology, there are three different choices listed in Figure 3.13. The leader keeps constant speed and changes its lateral position from 0 to 1 at time  $t = 15$ .

Since the speed dynamics is trivial, we just list the simulation results on the lateral dynamics. For random initial positions and orientations, Figure 3.15 shows how the lateral errors evolve with different  $\alpha$  values for double-graph control strategy. For  $\alpha = 0$ , vehicle formation with topology 1 is stable because it is acyclic. The top left plot shows that the formation is stable. However, the over-shoots of followers become larger and larger from vehicle 2 to vehicle 5. When  $\alpha$  increases, this accumulation disappears as shown in the second and third plots on the left. For the topology 2 and 3, when  $\alpha = 0$ , the formation is actually unstable due to the nonlinearity in vehicle dynamics and small stable margins. By increasing the value of  $\alpha$ , stable margins become larger and the effect of the nonlinearity becomes smaller. As shown in the central and right plots, the formation becomes stable.

Furthermore, we notice that, when  $\alpha$  is 0.5, vehicle formations with different topologies have almost same convergence behaviors shown in the bottom row in Figure 3.15. This can be explained due to the fact that, when critical points are moved into a small ellipse, the difference among these points generated by different topologies is negligible. Thus,

double-graph control strategy is robust to the interaction topology.

### 3.4.3 Experimental Results

In the experiments on the MVWT, vehicles are assigned to tracking a circle at certain speeds to overcome the space limitation. Figure 3.14 shows the result of a single-vehicle experiment. The center of the circle is at (3.0, 3.5), the radius is 1.6 meter and the reference speed is 0.8 m/sec.

The experiment for vehicle formation on the MVWT involves four vehicles. In the double-graph control strategy, vehicle 1 is the leader. Other vehicles try to follow the leader on the same circle trajectory with same speed. The topology for  $\mathcal{G}_1$  and interaction topology  $\mathcal{G}_2$  are shown in Figure 3.16. The graph coefficient  $\alpha$  is 0.5. The position offsets are defined by  $-0.6$  radian angle offset with respect to the circle trajectory.

According to Theorem 3.2.1, the whole formation is stable and has good performance because  $\|(1 - \alpha)H(s)\|_\infty = 0.5 < 1$ . We change the radius and speed of the leader and observe how each follower adjusts its movement. Figure 3.18 shows the radius changes and the speed changes of these four vehicles. The speeds of the followers are vibrating around the leader's speed and need much longer time to stabilize. This is because of the uncertainty in vehicle model, the noise introduced in the LPS, and the vulnerability of package loss in the wireless communication network (normally the package dropping rate is 7% – 10% for a four-vehicle formation). These experimental results verified the efficiency and robustness of the double-graph control strategy in critical environments.

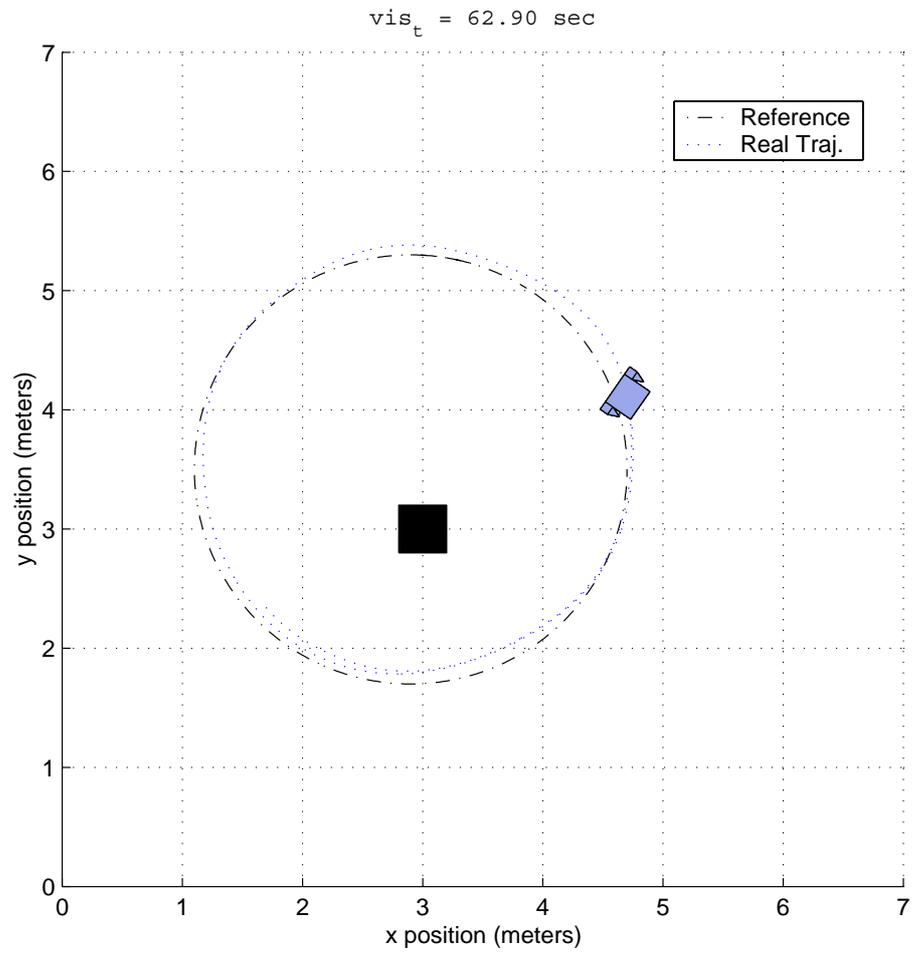


Figure 3.14: Top view of a single vehicle experiment

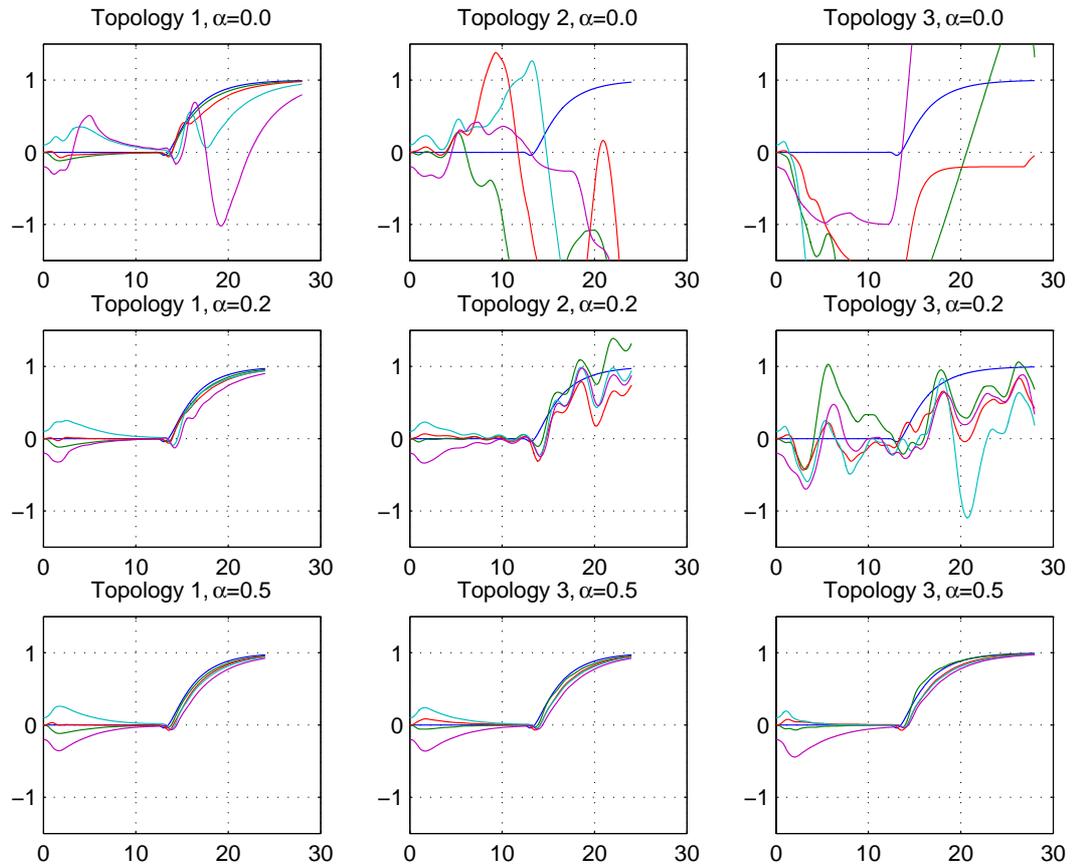


Figure 3.15: Lateral errors for vehicle formation with different parameters

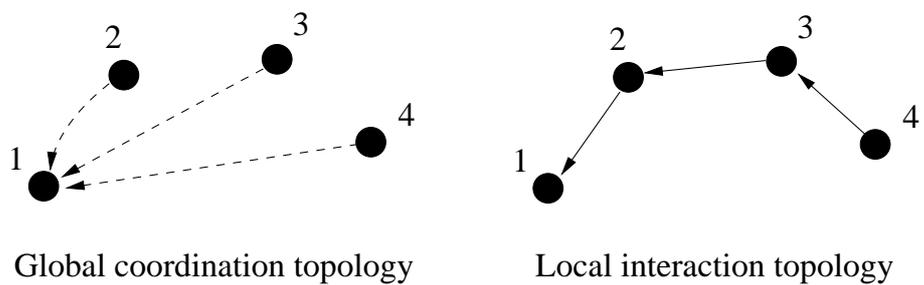


Figure 3.16: Topologies in double-graph control strategy of the MVWT experiment

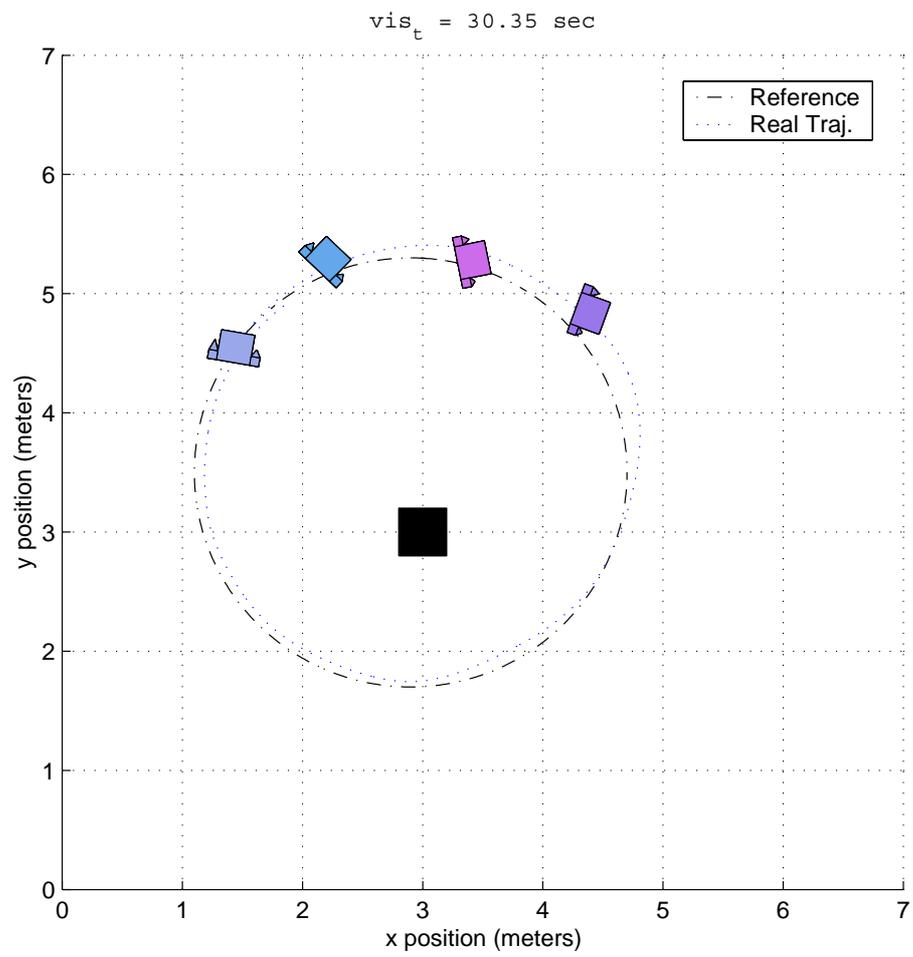


Figure 3.17: Top view of vehicle formation experiment

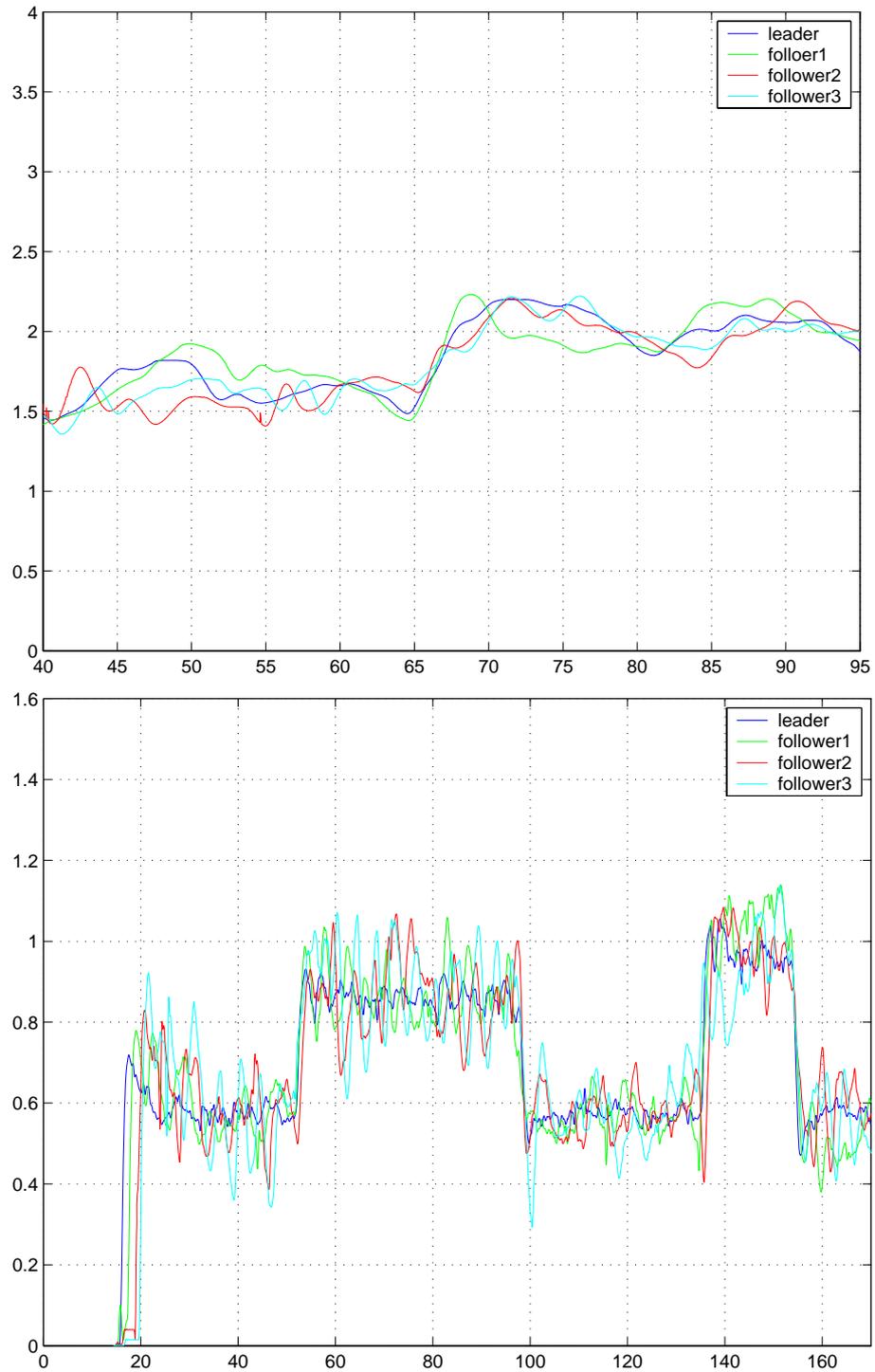


Figure 3.18: Vehicle formation experiment: radius and speed

## Chapter 4

# Consensus Protocols in Networked Multi-Agent Systems

One of the essential assumptions for double-graph control strategy of multi-agent systems is the global objective synchronization. In order to avoid centralized data collection and processing, distributed collective protocols over the communication network are needed where each agent updates its local copy of the global objective based on certain simple rules. One of the popular collective protocols is the consensus protocol. In this chapter, we focus on the continuous consensus protocol and discuss its convergence behavior based on the connectivity of the communication network. Also, a new type of consensus protocol is proposed and the performance tradeoffs are discussed.

### 4.1 Consensus Seeking for Multi-Agent Systems

Suppose in a networked multi-agent system,  $x_i$  denotes the local objective variable in agent  $i$ . A directed graph  $\mathcal{G}$  is used to represent the topology of the communication network. For each agent,  $x_i$  is updated according to the distributed rule

$$\dot{x}_i = - \sum_{j \in \mathcal{N}(i)} w_{ij} (x_i - x_j) \quad (4.1)$$

Table 4.1: Possible left eigenvectors for  $\mathcal{L}$  associated with  $\lambda(\mathcal{L}) = 0$ 

	No zero elements	Including zero elements
Uniform signs	type 1	type 2
Nonuniform signs	type 3	type 4

where  $w_{ij}$  is the positive weight associated with edge  $(v_i, v_j)$ . Thus, the dynamics of the synchronization process can be presented by

$$\dot{X} = -\mathcal{L}X \quad (4.2)$$

where  $X = [x_1, \dots, x_n]^T$  and  $\mathcal{L}$  is the Laplacian matrix. We say that the system in Equation (4.2) reaches a *consensus* if  $x_i = x_j$  for any  $i \neq j$ . This common value is called the *consensus state*, which is depicted by  $\eta$ .

**Lemma 4.1.1.** *Let  $\mathcal{G}$  be strongly connected. There exists a positive left eigenvector  $\mathbf{b} = [b_1, \dots, b_n]^T$  for  $\mathcal{L}$  associated with the eigenvalue zero.*

**Proof** Since 0 is a simple eigenvalue of  $\mathcal{L}$ , then there exists a left eigenvector  $\mathbf{b}$  so that  $\mathbf{b}^T \cdot \mathcal{L} = 0$ . Since  $-\mathbf{b}$  is also a left eigenvector, we actually need to prove that all elements in  $\mathbf{b}$  are nonzero and have the same signs. Table 4.1 lists the four possible types of  $\mathbf{b}$  and we will show that only type 1 is possible.

First, Let us prove that type 2 is impossible. Assume that there is only one zero element in  $\mathbf{b}$ . Without losing generality, we assume that  $b_n = 0$  and let  $\mathbf{l}_n$  denotes the last column of  $\mathcal{L}$ . According to the definition, off-diagonal elements of  $\mathcal{L}$  are nonnegative. So the first  $n - 1$  elements of  $\mathbf{l}_n$  must be zeros in order to satisfy the equation  $\mathbf{b}\mathbf{l}_n = 0$ . However, that means the in-degree of node  $v_n$  is zero, which violates the strongly connected definition. Then we assume that there are  $k$  zeros in  $\mathbf{b}$  where  $1 < k < n$ . Using a similar argument, we can show that there exists a subgraph composed of  $k$  nodes where there are no edges connecting other nodes to this subgraph. That is also impossible for a strongly connected graph.

Second, we prove that type 3 is impossible. Suppose there exist  $k$  negative elements and  $n - k$  positive elements in  $\mathbf{b}$ . By reindexing vertices, we can move all positive elements

in  $\mathbf{b}$  and get  $\mathbf{b} = [\mathbf{b}_+^T, \mathbf{b}_-^T]^T$  where  $\mathbf{b}_+$  is a positive vector with  $n - k$  elements and  $\mathbf{b}_-$  is a negative vector with  $k$  elements. We have

$$\mathbf{b}^T \cdot \mathcal{L} = [\mathbf{b}_+^T, \mathbf{b}_-^T] \begin{bmatrix} \mathcal{L}_{11} & \mathcal{L}_{12} \\ \mathcal{L}_{21} & \mathcal{L}_{22} \end{bmatrix} = [\mathbf{b}_+^T \cdot \mathcal{L}_{11} + \mathbf{b}_-^T \cdot \mathcal{L}_{21}, \mathbf{b}_+^T \cdot \mathcal{L}_{12} + \mathbf{b}_-^T \cdot \mathcal{L}_{22}] = \mathbf{0}_n^T.$$

It is obvious that  $\mathbf{b}_-^T \cdot \mathcal{L}_{21}$  has  $n - k$  nonpositive elements and  $\mathbf{b}_+^T \cdot \mathcal{L}_{12}$  has  $k$  nonnegative elements. Because  $\mathcal{G}$  is strongly connected, then the sum of all elements in  $\mathbf{b}_- \cdot \mathcal{L}_{21}$  is negative and the sum of all elements in  $\mathbf{b}_+ \cdot \mathcal{L}_{12}$  is positive since they cannot all be zeros. Moreover, the sum of all elements in  $\mathbf{b}_- \cdot \mathcal{L}_{21}$  and  $\mathbf{b}_- \cdot \mathcal{L}_{22}$  is zero since all rows of  $\mathcal{L}$  are zero-sum. Then the sum of elements in  $\mathbf{b}_- \cdot \mathcal{L}_{22}$  is positive. However, since the sum of all elements in  $\mathbf{b}_+ \cdot \mathcal{L}_{12}$  and  $\mathbf{b}_- \cdot \mathcal{L}_{22}$  should be zero, then the sum of elements in  $\mathbf{b}_- \cdot \mathcal{L}_{22}$  is negative. That results in a contradiction and  $\mathbf{b}$  cannot be type 3.

By combining previous arguments, it is easy to show that  $\mathbf{b}$  cannot be type 4 either.  $\square$

It is known that protocol (4.1) can achieve a consensus when the topology is strongly connected [64, 67]. However, it is unclear that where this consensus state is. The following theorem explicitly states the equilibrium point of the system (4.2).

**Theorem 4.1.1.** *Assuming  $\mathcal{G}$  is strongly connected, for any initial value  $X(0)$ , the final state of Equation (4.2) satisfies*

$$\lim_{t \rightarrow \infty} X = \eta \cdot \mathbf{1}_n$$

where

$$\eta = \frac{\mathbf{b}^T X(0)}{\sum_i b_i}.$$

**Proof** For system (4.2), the state vector  $X$  can be reorganized as

$$Z = V^{-1}X = \begin{bmatrix} b_1 & [b_2, \dots, b_n] \\ \mathbf{1}_{n-1} & -I_{n-1} \end{bmatrix} \cdot X = \begin{bmatrix} \sum b_i x_i \\ x_1 - x_2 \\ \vdots \\ x_1 - x_n \end{bmatrix}. \quad (4.3)$$

Suppose  $b_1 = 1$ , we present  $V$  explicitly based on the property of block matrices:

$$\begin{aligned}
V &= \begin{bmatrix} (\sum b_i)^{-1} & [b_2, \dots, b_n] \cdot (\mathbf{1}_{n-1} \cdot [b_2, \dots, b_n] + I_{n-1})^{-1} \\ (\sum b_i)^{-1} \mathbf{1}_{n-1} & -(\mathbf{1}_{n-1} \cdot [b_2, \dots, b_n] + I_{n-1})^{-1} \end{bmatrix} \\
&= (\sum b_i)^{-1} \begin{bmatrix} 1 & b_2 & \cdots & b_n \\ 1 & (b_2 - \sum b_i) & \cdots & b_n \\ \vdots & \vdots & \ddots & \vdots \\ 1 & b_2 & \cdots & (b_n - \sum b_i) \end{bmatrix}.
\end{aligned} \tag{4.4}$$

With little effort, we have

$$V^{-1} \mathcal{L} V = \begin{bmatrix} 0 & \mathbf{0}_{n-1} \\ \mathbf{0}_{n-1} & \hat{\mathcal{L}} \end{bmatrix} \tag{4.5}$$

where  $\hat{\mathcal{L}}$  is a full-rank  $(n-1)$ -by- $(n-1)$  matrix and all the eigenvalues of  $\hat{\mathcal{L}}$  are also eigenvalues of  $\mathcal{L}$ .

According to Chapter 2, all eigenvalues of  $\hat{\mathcal{L}}$  have positive real parts. Thus, all  $z_i$  except  $z_1$  converge to zero. This means that  $x_i$  converge to a common final value  $\eta$ . Because  $\dot{z}_1 = \sum b_i \dot{x}_i = 0$ , we have  $\sum b_i x_i(t) = \sum b_i \eta = \sum b_i x_i(0)$ , i.e.,

$$\eta = \frac{\sum_i b_i x_i(0)}{\sum_i b_i} = \frac{\mathbf{b}^T X(0)}{\sum_i b_i}.$$

□

This theorem gives a formula to calculate the final consensus value. If the directed graph  $\mathcal{G}$  is connected and symmetric,  $\mathcal{L}$  is symmetric and  $\mathbf{b} = \mathbf{1}_n$ . Thus,  $\eta = \sum x_i(0)/n$  and this synchronization process is called ‘‘average consensus seeking’’ because the final state of each agent is the average value of the initial states. A sufficient condition for average consensus seeking with the protocol in Equation (4.1) is that the directed graph  $\mathcal{G}$  is balanced, i.e., for any vertex, the in-degree equals the out-degree [64].

We list another lemma that is useful for consensus seeking over weakly connected graphs.

**Lemma 4.1.2.** For a strongly connected graph and  $P = \text{diag}([1, 0, 0, \dots])$ ,

$$\text{rank}(P + \mathcal{L}) = n.$$

**Proof** We use the same transformation as in Theorem 4.1.1 and get the following equation:

$$\begin{aligned} V^{-1}(P + \mathcal{L})V &= V^{-1}PV + V^{-1}\mathcal{L}V \\ &= \frac{1}{\sum b_i} \begin{bmatrix} b_1 & \cdots & b_n \\ \vdots & \vdots & \vdots \\ b_1 & \cdots & b_n \end{bmatrix} + \begin{bmatrix} 0 & \mathbf{0} \\ \mathbf{0} & \hat{\mathcal{L}}_{22} \end{bmatrix}. \end{aligned} \quad (4.6)$$

The rank of the first matrix is 1 and the rank of the second matrix is  $n - 1$ . However,  $[b_1, \dots, b_n]$  is linear independent of the row space of the second matrix since  $b_1 \neq 0$ . So  $P + \mathcal{L}$  is fullrank.  $\square$

For other diagonal matrix  $P$  with diagonal elements are nonnegative and at least one is nonzero, it can be shown that  $\mathcal{L} + P$  is full rank based on Lemma 4.1.2.

When the graph is not strongly connected, the protocol cannot guarantee that all  $x_i$  converges to a consensus from any initial conditions. However, Reducing the graph based on strongly connected components can give us more insights.

**Lemma 4.1.3.** If  $\mathcal{G}$  is weakly connected with  $q$  root strong components,  $\text{rank}(\mathcal{L}) = N - q$ .

**Proof** First of all,  $\mathcal{L}$  is reducible since  $\mathcal{G}$  is weakly connected. Suppose there are  $u$  strong components in  $\mathcal{G}$ , then  $\mathcal{L}$  can be described in the following form when we change the vertex index properly:

$$\mathcal{L} = \begin{bmatrix} l_1 & & & & & & \\ & \ddots & & & & & \\ & & l_q & & & & \\ * & * & * & l_{q+1} & & & \\ * & * & * & * & \ddots & & \\ * & * & * & * & * & l_u & \end{bmatrix}. \quad (4.7)$$

Thus,

$$\text{rank}(\mathcal{L}) = \sum_i \text{rank}(l_i).$$

The first  $q$  diagonal blocks are Laplacian matrices of the  $q$  root strong components. For the last  $u - q$  diagonal blocks, any one of them can be written as the linear combination of a Laplacian matrix of the corresponding strong component and a diagonal matrix with non-negative diagonal elements. According to Lemma 4.1.2, they are full rank and the result follows.  $\square$

**Theorem 4.1.2.** *Protocol (4.1) can reach a consensus from any initial value if and only if  $\mathcal{G}$  has a unique root strong component, i.e.,  $\text{rank}(\mathcal{L}) = n - 1$ .*

**Proof** When  $\mathcal{G}$  has a unique root strong component, it can be reduced to a root directed tree and  $\mathcal{L}$  can be written as a block lower triangular matrix as in Equation (4.7) where  $l_1$  is the Laplacian matrix of the root strong component. Any other diagonal block  $l_i$  is fullrank and all the eigenvalues have positive real parts. So  $\text{rank}(\mathcal{L}) = n - 1$  and 0 is a simple eigenvalue.

suppose the root strong component has  $n_1$  vertices and  $[b_1, \dots, b_{n_1}]$  is a left eigenvector of  $l_1$  associated with zero eigenvalue. Using the same trick as in Theorem 4.1.1, the state vector  $X$  can be reorganized as

$$Z = V^{-1}X = \begin{bmatrix} b_1 & [b_2, \dots, b_{n_1}] & 0 \\ \mathbf{1}_{n_1-1} & -I_{n_1-1} & 0 \\ \mathbf{1}_{n-n_1} & 0 & -I_{n-n_1} \end{bmatrix} \cdot X = \begin{bmatrix} \sum_{i=1}^{n_1} b_i x_i \\ x_1 - x_2 \\ \vdots \\ x_1 - x_n \end{bmatrix} \quad (4.8)$$

and we have

$$\dot{Z} = -V^{-1}\mathcal{L}VZ, \quad (4.9)$$

where

$$V^{-1}\mathcal{L}V = \begin{bmatrix} 0 & \mathbf{0} \\ \mathbf{0} & \hat{\mathcal{L}} \end{bmatrix}. \quad (4.10)$$

Since  $-\hat{\mathcal{L}}$  is stable, the system converges to a consensus and

$$\eta = \sum_{i=1}^{n_1} b_i x_i(0) / \sum_{i=1}^{n_1} b_i.$$

□

## 4.2 Multi-Hop Replay Protocols for Consensus Seeking

In this section, we discuss the convergence speed for average consensus seeking when  $\mathcal{G}$  is connected and symmetric. Since  $\mathcal{G}$  is symmetric, eigenvalues of  $\mathcal{L}$  are real and denoted by  $0 = \lambda_1(\mathcal{L}) < \lambda_2(\mathcal{L}) \leq \dots \leq \lambda_n(\mathcal{L})$ . Thus, the convergence speed is bounded by the second smallest eigenvalue  $\lambda_2(\mathcal{L})$  that is called “algebraic connectivity” [101]. Intuitively, the more links there are in the graph, the bigger the algebraic connectivity is, and the faster the convergence should be. An upper bound of the convergence speed can be found for the consensus protocol (4.1).

**Lemma 4.2.1.** *The maximum value of the second-smallest eigenvalue  $\lambda_2$  of  $\mathcal{L}$  for a symmetric and connected directed graph  $\mathcal{G}$  with  $n$  vertices is  $\sum_{i \neq j} w_{ij} / (n - 1)$ .*

**Proof** For  $\mathcal{L}$ , we have

$$(n - 1)\lambda_2(\mathcal{L}) \leq \sum_{i=1}^n \lambda_i(\mathcal{L}) = \text{tr}(\mathcal{L}) \leq \sum_{i \neq j} w_{ij}$$

where  $\text{tr}(\mathcal{L})$  is the trace of  $\mathcal{L}$ . Thus,

$$\lambda_2(\mathcal{L}) \leq \sum_{i \neq j} w_{ij} / (n - 1).$$

□

The equation holds only when  $\mathcal{G}$  is complete and all weights  $w_{ij}$  are identical. Thus, consensus protocol (4.1) reaches its maximum convergence speed if we configure the topology to be a complete graph with uniform weights.

Many applications of consensus protocols, such as peer-to-peer networks, sensor networks, and distributed Kalman filtering, require fast converge speeds. In [109], a consensus process over a symmetric network is treated as an optimal linear iteration problem and the convergence speed can be increased by finding the optimal weights associated with each edge. But the global structure of the network must be known beforehand. In [73], a “random rewiring” procedure is proposed to boost the convergence speed for large scale graphs. However, in many applications, physically changing the topology is difficult. A new type of consensus protocol is introduced in following subsections in order to get a better convergence speed without changing the topology and edge weights.

### 4.2.1 Two-Hop Relay Protocol

The distributed two-hop relay protocol is described as

$$\dot{x}_i = - \sum_{j \in N(i)} w_{ij} (x_i - x_j) + \sum_{k \in N(j)} w_{jk} (x_i - x_k). \quad (4.11)$$

In this protocol, what vertex  $v_j$  sends to  $v_i$  is not only its own state value, but also a collection of its instantaneous neighbors’ states. It is equivalent to adding virtual “two-hop” paths to the original graph  $\mathcal{G}$ . We define the *two-hop directed graph*  $\hat{\mathcal{G}} = (\mathcal{V}, \hat{\mathcal{E}})$  as the graph that has the same vertex set, and all the edges are “two-hop” paths of  $\mathcal{G}$ . Figure 4.1 shows an example of the two-hop directed graph.



Figure 4.1: A directed graph and its two-hop directed graph

There may exist self-loops in two-hop graph  $\hat{\mathcal{G}}$ , i.e., the head and tail of an edge are

the same. This is common when  $\mathcal{G}$  is symmetric. However, according to the relay protocol (4.11), these self-loops have no contributions to the dynamics. So, these self-loops are omitted. Moreover, multiple two-hop paths may exist between a pair of vertices. In  $\hat{\mathcal{G}}$ , we consider them as one edge and its weight equals the sum of the two-hop paths. Thus, the adjacency matrix  $\hat{\mathcal{A}} = \{\hat{a}_{ik}\}$  of  $\hat{\mathcal{G}}$  is

$$\hat{a}_{ik} = \begin{cases} \sum_j w_{ij}w_{jk}, & (v_i, v_k) \in \hat{\mathcal{E}} \\ 0, & \text{otherwise.} \end{cases}$$

The corresponding out-degree diagonal and Laplacian matrices in  $\hat{\mathcal{G}}$  are denoted by  $\hat{D}$  and  $\hat{\mathcal{L}}$ , respectively. For a directed graph  $\mathcal{G}$  with two-hop relay protocol (4.11), consider the joint graph  $\tilde{\mathcal{G}} = \mathcal{G} \cup \hat{\mathcal{G}} = (\mathcal{V}, \mathcal{E} \cup \hat{\mathcal{E}})$ , and it is obvious that the two-hop relay protocol is a consensus protocol of  $\tilde{\mathcal{G}}$ . Thus, the dynamics of the whole system is described as

$$\dot{X} = -\tilde{\mathcal{L}}X \quad (4.12)$$

where  $\tilde{\mathcal{L}} = \mathcal{L} + \hat{\mathcal{L}}$ .

The two-hop relay protocol needs extra communication bandwidth. Assuming that the graph is static, we rewrite the protocol (4.1) as

$$\dot{x}_i = -x_i \sum_{j \in \mathcal{N}(i)} w_{ij} + \sum_{j \in \mathcal{N}(i)} w_{ij} x_j \quad (4.13)$$

and the two-hop relay protocol (4.11) as

$$\begin{aligned} \dot{x}_i = & -x_i \sum_{j \in \mathcal{N}(i)} w_{ij} (1 + \sum_{k \in \mathcal{N}(j)} w_{jk}) \\ & + \sum_{j \in \mathcal{N}(i)} w_{ij} (x_j + \sum_{k \in \mathcal{N}(j)} w_{jk} x_k). \end{aligned} \quad (4.14)$$

For protocol (4.1), what link  $(v_i, v_j)$  transmits is the value of  $x_j$ . For protocol (4.11), what link  $(v_i, v_j)$  transmits is the value of  $x_j$ ,  $\sum w_{jk} x_k$ , and  $\sum w_{jk}$ . However, for a static graph,  $\sum w_{jk}$  is constant and only needs to be transmitted once. Thus, the two-hop relay protocol needs twice the communication bandwidth as protocol (4.1) needs except at the very beginning.

## 4.2.2 Convergence Speed of Two-Hop Relay Protocol

When  $\mathcal{G}$  is connected and symmetric, the two-hop graph  $\hat{\mathcal{G}}$  is symmetric, and the joint graph  $\tilde{\mathcal{G}}$  is also symmetric and connected.

**Theorem 4.2.1.** *If the original graph  $\mathcal{G}$  is connected and symmetric, then*

$$\lambda_2(\mathcal{L}) \leq \lambda_2(\tilde{\mathcal{L}}). \quad (4.15)$$

**Proof** For any vector  $x$ , it is true that

$$\begin{aligned} x^T \tilde{\mathcal{L}}x &= x^T \mathcal{L}x + x^T \hat{\mathcal{L}}x \\ &= \sum_{(i,j) \in \mathcal{E}} w_{ij}^2 (x_i - x_j)^2 \\ &\quad + \sum_{(i,j) \in \hat{\mathcal{E}}} w_{ij}^2 (x_i - x_j)^2. \end{aligned}$$

We pick a unit vector  $x$  and let it be orthogonal to  $\mathbf{1}_n$ , then

$$\frac{x^T \mathcal{L}x}{x^T x} = \frac{\sum_{(v_i, v_j) \in \mathcal{E}} (x_i - x_j)^2}{\sum_{v_i \in \mathcal{V}} x_i^2} \geq \lambda_2(\mathcal{L})$$

and the equation holds only when  $x$  is an eigenvector associated with  $\lambda_2(\mathcal{L})$ .

Combining these two results, if we take  $x$  to be a unit eigenvector of  $\tilde{\mathcal{L}}$ , orthogonal to  $\mathbf{1}_n$ , associated with eigenvalue  $\lambda_2(\tilde{\mathcal{L}})$ , then we have

$$\lambda_2(\tilde{\mathcal{L}}) = \frac{x^T \tilde{\mathcal{L}}x}{x^T x} = \frac{x^T (\mathcal{L} + \hat{\mathcal{L}})x}{x^T x} \geq \lambda_2(\mathcal{L}) + \frac{x^T \hat{\mathcal{L}}x}{x^T x}. \quad \square \quad (4.16)$$

Theorem 4.1.1 shows that the two-hop relay protocol increases the convergence speed. The improvement depends on the topology of  $\hat{\mathcal{G}}$ . It can be shown that the edge set of  $\hat{\mathcal{G}}$  is not empty if the original graph has more than two vertices. Furthermore, if  $\hat{\mathcal{G}}$  is connected, we have

$$\lambda_2(\tilde{\mathcal{L}}) \geq \lambda_2(\mathcal{L}) + \lambda_2(\hat{\mathcal{L}}). \quad (4.17)$$

since  $x^T \hat{\mathcal{L}}x / x^T x \geq \lambda_2(\hat{\mathcal{L}}) > 0$ . Thus, two-hop relay protocol improves the algebraic connectivity at least by  $\lambda_2(\hat{\mathcal{L}})$ .

However, it is not true that  $\hat{\mathcal{G}}$  is always connected. Figure 4.2 shows a simple example. The original graph on the left is symmetric and connected, but the two-hop graph on the right is composed of two disconnected subgraphs.

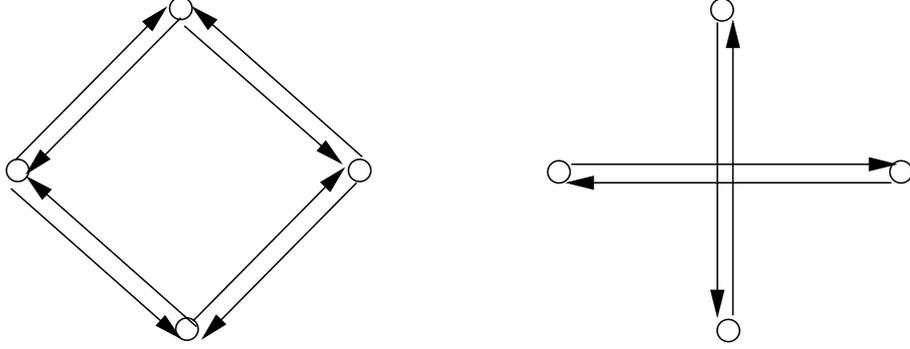


Figure 4.2: An example of disconnected two-hop directed graph

### 4.2.3 Multi-Hop Relay Protocol

It is possible to extend the two-hop relay protocol to the multi-hop relay protocol. The protocol for  $m$ -hop relay protocol can be written as

$$\dot{x}_i = - \underbrace{\sum_j w_{ij}((x_i - x_j) + \sum_k w_{jk}((x_i - x_k) + \dots))}_{m \text{ layers}}. \quad (4.18)$$

Clearly it adds more virtual edges to the original graph and enforces the convergence speed. There are three drawbacks. First, the worst case computation complexity of the  $m$ -hop relay protocol on each agent is  $O(n^{m-1})$ . For large scale networks, it is not scalable and quickly becomes infeasible as  $m$  increases. Second,  $m$ -times communication bandwidth are needed. Third, communication delays will accumulate along  $m$ -hop paths and that makes the protocol very sensitive to communication latency.

## 4.3 Two-Hop Relay Protocols with Time Delays

For communication delays, we consider the transfer function associated with edge  $(v_i, v_j)$  with latency  $\tau_{ij}$  is  $e^{-\tau_{ij}s}$ . Delays will be accumulated along two-hop paths. We

study the simplest case where all delays are identical, i.e.,  $\tau_{ij} = \tau$  for any  $(v_i, v_j) \in \mathcal{E}$ , and the graph  $\mathcal{G}$  is connected and symmetric with uniform edge weights. The protocol (4.1) can be written as

$$\dot{x}_i = - \sum_{j \in \mathcal{N}(i)} w_{ij} (x_i(t - \tau) - x_j(t - \tau)) \quad (4.19)$$

and the two-hop relay protocol is

$$\begin{aligned} \dot{x}_i = & - \sum_{j \in \mathcal{N}(i)} w_{ij} \left( (x_i(t - \tau) - x_j(t - \tau)) \right. \\ & \left. + \sum_{k \in \mathcal{N}(j)} w_{jk} (x_i(t - 2\tau) - x_k(t - 2\tau)) \right). \end{aligned} \quad (4.20)$$

Equation (4.2) and (4.12) change to

$$\dot{X} = -\mathcal{L} \cdot X(t - \tau) \quad (4.21)$$

and

$$\dot{X} = -\mathcal{L} \cdot X(t - \tau) - \hat{\mathcal{L}} \cdot X(t - 2\tau), \quad (4.22)$$

respectively.

Let  $Z = V^{-1}X$  where

$$V^{-1} = \begin{bmatrix} 1 & \mathbf{1} \\ \mathbf{1} & -I \end{bmatrix}. \quad (4.23)$$

For the two-hop relay protocol, we have to

$$\dot{Z} = -V^{-1} \mathcal{L} V Z(t - \tau) - V^{-1} \hat{\mathcal{L}} V Z(t - 2\tau). \quad (4.24)$$

Note that

$$V^{-1} \mathcal{L} V = \begin{bmatrix} 0 & \mathbf{0}_{n-1} \\ \mathbf{0}_{n-1} & \mathcal{L}_{22} \end{bmatrix} \text{ and } V^{-1} \hat{\mathcal{L}} V = \begin{bmatrix} 0 & \mathbf{0}_{n-1} \\ \mathbf{0}_{n-1} & \hat{\mathcal{L}}_{22} \end{bmatrix}.$$

Assume that  $X(t) = 0$  for any  $t < 0$ . Taking the Laplace transform of Equation (4.24), all states except  $z_1$  of this autonomous system asymptotically converge to 0, i.e.,  $x_i$  converges

to  $\eta = \sum x_i(0)/n$ , if and only if the following characteristic polynomial

$$p_{22}(s, e^{-\tau s}) = \det(sI + \mathcal{L}e^{-\tau s} + \hat{\mathcal{L}}e^{-2\tau s}) \quad (4.25)$$

has no zero in the closed right half plane (RHP). This condition is equivalent to the case that the characteristic polynomial

$$p(s, e^{-\tau s}) = \det(sI + \mathcal{L}e^{-\tau s} + \hat{\mathcal{L}}e^{-2\tau s}) \quad (4.26)$$

has no zero in the closed RHP except the simple zero at the origin. In [110],  $p_{22}(s, e^{-\tau s})$  and  $p(s, e^{-\tau s})$  are called real *quasipolynomials* of  $s$ . In the rest of this chapter, we will consistently use this name. A similar quasipolynomial for (4.21) is

$$p(s, e^{-\tau s}) = \det(sI + \mathcal{L}e^{-\tau s}). \quad (4.27)$$

One essential property of quasipolynomials is the continuity of the zeros with respect to time delay. In other words, when  $\tau$  increases, zeros in the left half plane (LHP) move to RHP. Time delay does not affect the zero  $s = 0$ . We need to find the minimum value of  $\tau$  such that the first stable zero crosses the imaginary axis. Besides, the conjugate symmetry property of quasipolynomials makes it possible to calculate the critical value of the time delay and the corresponding crossing frequency.

**Definition 4.3.1.** *Given initial value  $X(0)$  and assumption  $X(t) = 0$  for  $t < 0$ , the smallest value of  $\tau$  such that the system cannot converge to a consensus is determined as*

$$\tau^* = \min\{\tau > 0 \mid p(j\omega, e^{-j\tau\omega}) = 0 \text{ and } \omega \neq 0\}, \quad (4.28)$$

*which is called the delay margin of the consensus protocol.*

It is true that, for any  $\tau \in [0, \tau^*)$ , the system of (4.21) or (4.22) converges to an average consensus.

**Lemma 4.3.1.** *Let  $\tau^*$  and  $\tilde{\tau}^*$  indicate the delay margin of (4.21) and (4.22), respectively. Then  $\tau^* \geq \tilde{\tau}^*$ .*

**Proof** First, let us find  $\tau^*$ . Based on Schur theorem, there exists a unitary matrix  $T$  such that  $U = T^{-1}\mathcal{L}T$  is upper triangular with the eigenvalues along the diagonal. The quasipolynomial (4.27) is

$$\begin{aligned}\det(sI + \mathcal{L}e^{-\tau s}) &= \det(T(sI + Ue^{-\tau s})T^{-1}) \\ &= s \cdot \prod_{i=2}^n (s + \lambda_i(\mathcal{L})e^{-\tau s}).\end{aligned}$$

We need to find the smallest  $\tau > 0$  such that the first stable zero reaches the imaginary axis. Let  $s = j\omega$  and we have

$$j\omega = -e^{\tau j\omega} \lambda_i(\mathcal{L}). \quad (4.29)$$

Solving this equation gives us

$$\begin{cases} \omega = \lambda_i(\mathcal{L}) \neq 0 \\ \tau = \pi/2\lambda_i(\mathcal{L}). \end{cases} \quad (4.30)$$

So the delay margin

$$\tau^* = \min \frac{\pi}{2\lambda_i(\mathcal{L})} = \pi/2\lambda_n(\mathcal{L}). \quad (4.31)$$

Next, we consider  $\tilde{\tau}^*$ . The above approach fails for the quasipolynomial  $\det(sI + \mathcal{L}e^{-\tau s} + \hat{\mathcal{L}}e^{-2\tau s})$ . However, it is obvious that  $\tilde{\tau}^*$  should be no bigger than the delay margin for  $\det(sI + (\mathcal{L} + \hat{\mathcal{L}})e^{-\tau s})$ , which is  $\pi/2\lambda_n(\tilde{\mathcal{L}})$ . Moreover,  $\lambda_n(\tilde{\mathcal{L}}) \geq \lambda_n(\mathcal{L})$  according to [101]. So we have

$$\tau^* = \pi/(2\lambda_n(\mathcal{L})) \geq \pi/(2\lambda_n(\tilde{\mathcal{L}})) \geq \tilde{\tau}^*. \quad \square$$

This lemma just shows us that the delay sensitivity of the two-hop relay protocol is no better than protocol (4.1). The following theorem gives us explicit results on  $\tilde{\tau}^*$  by using the frequency-sweeping method.

**Theorem 4.3.1.** *For system (4.22), define*

$$\bar{\tau}_i = \min_{1 \leq k \leq n-1} \theta_k^i / \omega_k^i$$

when the generalized eigenvalues  $\lambda_i(G(s), H)$  satisfy the following equation:

$$\lambda_i(G(j\omega_k^i), H) = e^{-j\theta_k^i}$$

for some  $\omega_k^i \in (0, \infty)$  and  $\theta_k^i \in [0, 2\pi)$ , where

$$G(s) = \begin{bmatrix} 0 & I \\ -sI & -\mathcal{L}_{22} \end{bmatrix} \text{ and } H = \begin{bmatrix} I & 0 \\ 0 & \hat{\mathcal{L}}_{22} \end{bmatrix}.$$

Then the consensus delay margin of (4.22) is

$$\tilde{\tau}^* = \min_{1 \leq i \leq 2(n-1)} \bar{\tau}_i.$$

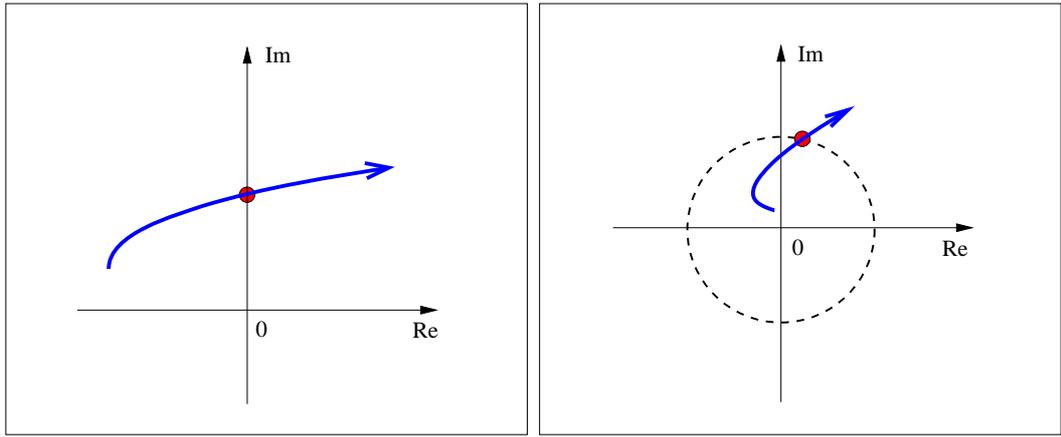


Figure 4.3: Locus of the zero of the quasipolynomial and the generalized eigenvalue

**Proof** *Generalized eigenvalue* for matrix pair  $(A, B)$  is defined as  $\lambda(A, B)$  that satisfies  $Ay = \lambda(A, B) \cdot By$  for a nonzero vector  $y$ . The vector  $y$  is called the *generalized eigenvector*. When  $B = I$ ,  $\lambda(A, B) = \lambda(A)$ . It is a well-known fact that the number of finite generalized eigenvalues for  $(A, B)$  is at most equal to  $\text{rank}(B)$ . Also, if  $\text{rank}(B)$  is constant,  $\lambda(A, B)$  is continuous with respect to the elements of  $A$ .

Based on the aforementioned similarity transform, system (4.22) converges to a con-

sensus if the following system

$$\begin{bmatrix} \dot{z}_2 \\ \vdots \\ \dot{z}_n \end{bmatrix} = -\mathcal{L}_{22} \begin{bmatrix} z_2(t-\tau) \\ \vdots \\ z_n(t-\tau) \end{bmatrix} - \hat{\mathcal{L}}_{22} \begin{bmatrix} z_2(t-2\tau) \\ \vdots \\ z_n(t-2\tau) \end{bmatrix} \quad (4.32)$$

is stable. According to the Schur determinant complement,

$$\det \left( \begin{bmatrix} A & B \\ C & D \end{bmatrix} \right) = \det(A) \cdot \det(D - CA^{-1}B).$$

Thus, we have

$$\begin{aligned} \det(sI + \mathcal{L}_{22}e^{-\tau s} + \hat{\mathcal{L}}_{22}e^{-2\tau s}) &= \det(e^{-\tau s}I) \det(\hat{\mathcal{L}}_{22}e^{-\tau s} + \mathcal{L}_{22} + sI \cdot e^{\tau s}) \\ &= \det \left( \begin{bmatrix} I \cdot e^{-\tau s} & -I \\ sI & \hat{\mathcal{L}}_{22}e^{-\tau s} + \mathcal{L}_{22} \end{bmatrix} \right) \\ &= \det \left( e^{-\tau s} \begin{bmatrix} I & \\ & \hat{\mathcal{L}}_{22} \end{bmatrix} - \begin{bmatrix} 0 & I \\ -sI & -\mathcal{L}_{22} \end{bmatrix} \right). \end{aligned} \quad (4.33)$$

Now the quasipolynomial with two delay terms transfers to a new quasipolynomial with a single delay term as

$$\det(sI + \mathcal{L}_{22}e^{-\tau s} + \hat{\mathcal{L}}_{22}e^{-2\tau s}) = (-1)^{2n-2} \det(G(s) - e^{-\tau s}H) \quad (4.34)$$

where

$$G(s) = \begin{bmatrix} 0 & I \\ -sI & -\mathcal{L}_{22} \end{bmatrix} \text{ and } H = \begin{bmatrix} I & 0 \\ 0 & \hat{\mathcal{L}}_{22} \end{bmatrix}. \quad (4.35)$$

Since  $\tau \in \mathcal{R}$ , whenever a zero is located on the imaginary axis, there exists  $s = j\omega$  so that  $e^{-j\omega\tau}$  is a generalized eigenvalue of  $(G(s), H)$  with magnitude 1. Figure 4.3 shows this correspondence by plotting the zero locus of the quasipolynomial and the eigenvalue locus. Thus, we can transfer the problem of finding a  $\tau$  so that the quasipolynomial has zeros with pure imaginary parts to the problem of finding a  $\omega$  so that  $(G(s), H)$  has a generalized

eigenvalue with magnitude 1.

Since  $\text{rank}(H) = 2(n-1)$ , there are at most  $2(n-1)$  generalized eigenvalues of  $(G(s), H)$ . When  $s$  moves along the imaginary axis from 0 to  $j\infty$ , there exists at most  $n-1$  frequency  $\omega_k^i$  so that  $\|\lambda_i(G(j\omega_k^i), H)\|_2 = \|e^{-j\theta_k^i}\|_2 = 1$ . So the delay margin  $\tilde{\tau}^*$  is the minimum value of all possible  $\tilde{\tau}_k^i = \theta_k^i/\omega_k^i$ .  $\square$

Because there exist many efficient algorithms for generalized eigenvalue searching, it is much easier for us to find  $\lambda(G(s), H)$  than to solve the quasipolynomial. However, for large scale graphs, it is still a difficult problem due to the sizes of  $G$  and  $H$ .

## 4.4 Examples and Simulation Results

In order to verify the efficiency of the two-hop relay consensus protocol, we test it on three different topologies in Figure 4.4, denoted as  $\mathcal{G}_1$ ,  $\mathcal{G}_2$ , and  $\mathcal{G}_3$  from left to right. Topology  $\mathcal{G}_1$  is a 2-regular graph,  $\mathcal{G}_2$  is a net in which each vertex connects to the vertices located inside a certain range, and  $\mathcal{G}_3$  is a complete graph. All of them have ten vertices. They are all symmetric and connected. Each pair of edges with same vertices, i.e.,  $(v_i, v_j)$  and  $(v_j, v_i)$ , is denoted by a single link. For simplicity, we assume that  $w_{ij} = 1$  for any edge.

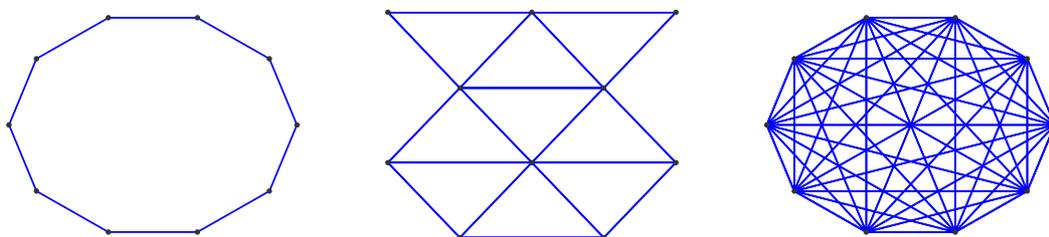


Figure 4.4: Three different topologies:  $\mathcal{G}_1$ ,  $\mathcal{G}_2$ , and  $\mathcal{G}_3$

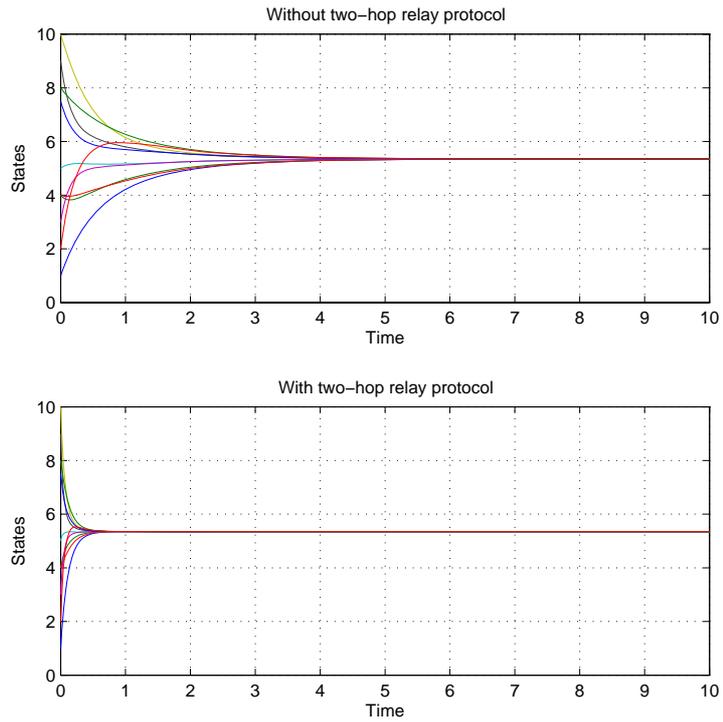
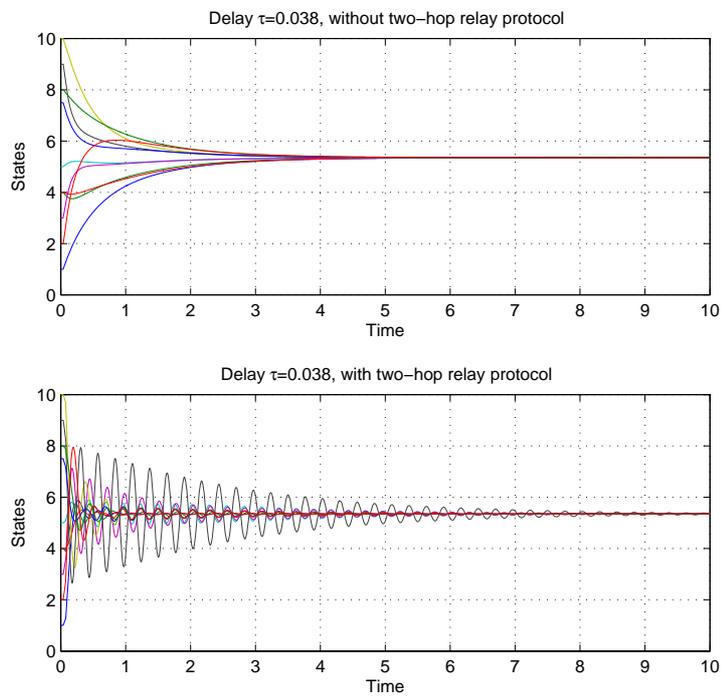
Figure 4.5 to Figure 4.8 show the simulation results of  $\mathcal{G}_2$  with the same initial conditions and different delays. Note that, even though the system can become unstable, the sum of the states keeps constant. Table 4.2 shows the algebraic connectivities and delay margins for all three graphs with or without two-hop relay protocols. Delay margins without relays are calculated according to Equation (4.31). Delay margins for two-hop relay protocols are computed using frequency sweep method mentioned in Theorem 4.3.1. Note that the mag-

Table 4.2: Performance vs. robustness for relay protocols

	<b>Algebraic connectivity <math>\lambda_2</math></b>		<b>Delay margin <math>\tau^*</math></b>	
	Without relay	With relay	Without relay	With relay
$\mathcal{G}_1$	0.382	1.7639	0.3927	0.1796
$\mathcal{G}_2$	0.9118	7.3846	0.2167	0.0396
$\mathcal{G}_3$	10	90	0.1571	0.0095

nitudes of generalized eigenvalues inevitably exceed 1 after a certain  $\omega$ . The computation needs to be done only over a finite frequency interval. We actually run the computation twice. The first time we try to find an appropriate frequency interval using big frequency steps. The second time we use much smaller frequency steps over the interval in order to find an accurate value of the delay margin.

For each graph, the relay protocol improves the convergence speed. However, time-delay robustness is impaired due to the delay accumulation along two-hop paths. Moreover, along the columns of the table, we can tell that algebraic connectivity increases and delay margin decreases when the graph includes more links. We put these data in Figure 4.9. For each bar, the right lower point corresponds to protocol (4.1) and the left upper point corresponds to the two-hop relay protocol. It is true that relay protocols actually boost convergence speed by increasing the algebraic connectivity and sacrificing the delay margin.

Figure 4.5: States of graph  $\mathcal{G}_2$  with no delayFigure 4.6: States of graph  $\mathcal{G}_2$  with delay  $\tau = 0.038$

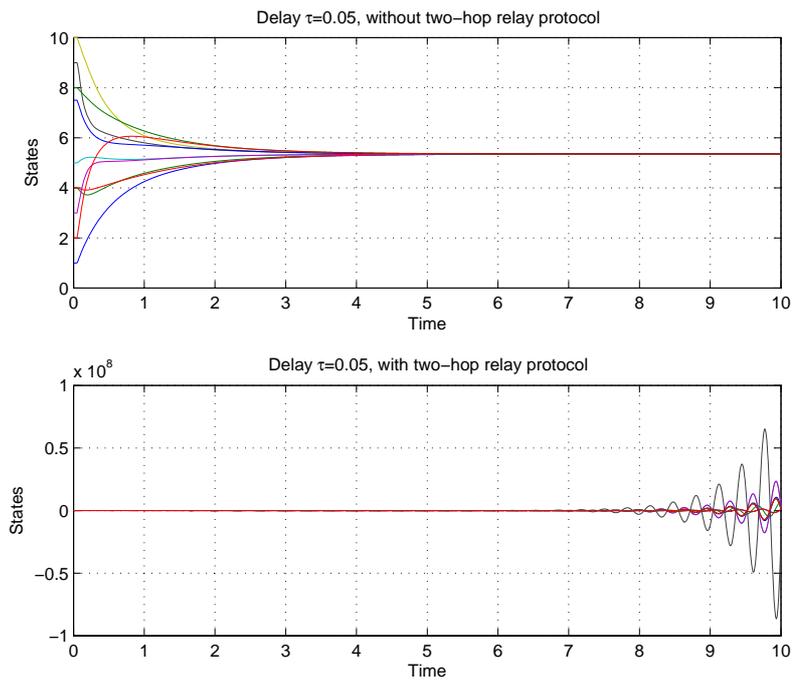


Figure 4.7: States of graph  $\mathcal{G}_2$  with delay  $\tau = 0.05$

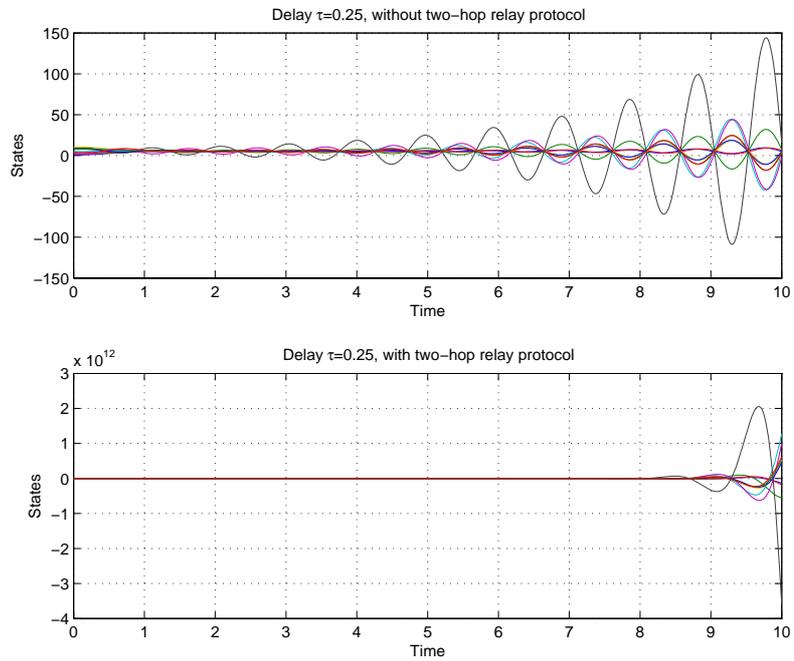


Figure 4.8: States of graph  $\mathcal{G}_2$  with delay  $\tau = 0.25$

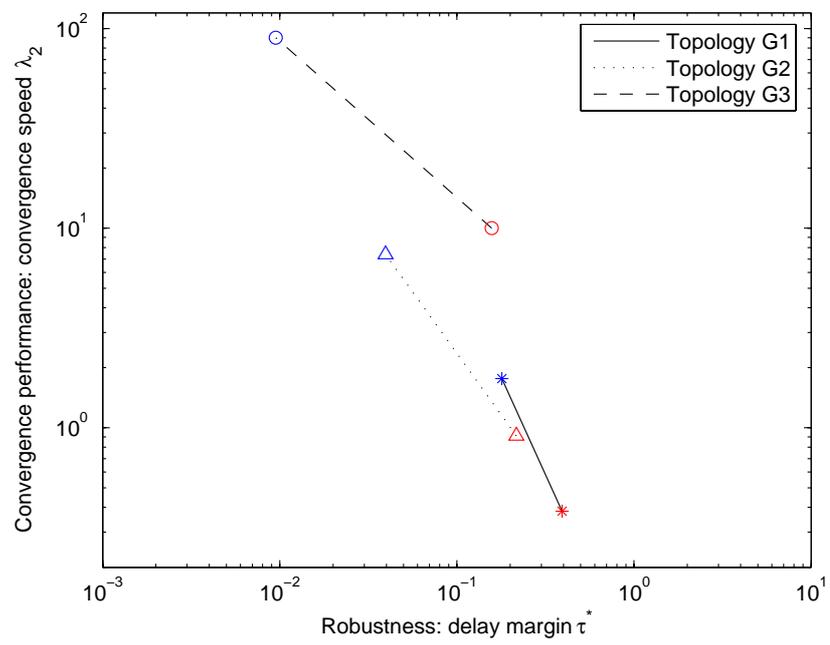


Figure 4.9: Tradeoff between  $\lambda_2$  and  $\tau^*$

## Chapter 5

# Packet-Based State Estimation Using Multiple Description Codes

In conventional model-based state estimation theory, one standard assumption is that observation data can be transmitted from observer to state estimator continually. The data may be corrupted by noise, but it is transmitted with infinite precision. That means we need reliable communication channels with infinite bandwidth. However, all practical communication networks have limited channel capacities and data packets may be delayed or dropped during the transmission. There is an increasing attention in the systems and control community on studying the effects of finite bandwidth and stochastic packet losses on networked control systems (NCS). In this chapter, we investigate the following questions: how does the unreliability of the communication network affect state estimation and what can we do to compensate for this unreliability?

Most modern communication networks use packet-based protocols to transfer data. Transmission control protocol (TCP) is one of the standard protocols that controls retransmission of lost packets and guarantees reliable delivery. In TCP, the module at the far end sends back an acknowledgement for packets that have been successfully received. A timer at the sending module causes a timeout if an acknowledgement is not received within a reasonable round-trip time and the packet will then be retransmitted. For state estimation problem, this retransmission scheme faces challenges to satisfy the real-time demands because of the unpredictable large transmission delays. User datagram protocol (UDP) is another common protocol that is faster and more efficient for time-sensitive pur-

poses because it does not retransmit lost packets [111]. In this chapter, we assume that the communication network delivers data packets based on an “UDP-like” mechanism and we focus on the packet-based state estimation problem with the presence of random packet drops. This work has immediate applications in the aforementioned networked multi-agent systems where it is important that each agent be able to estimate the states of others.

We focus on investigating the convergence behavior of discrete-time Kalman filters. Multiple description (MD) codes, a type of network source code, are used to compensate for packet loss. MD codes have been studied in information theory for over 30 years and successfully used in transmission of real-time speech and audio/video signals. The efficiency of MD codes has been proved in situations where data can be used at various resolution levels. However, this is the first time that MD codes are used for the state estimation problem. As mentioned in Chapter 2, packet drops are described by two mathematic models, the i.i.d. Bernoulli model and the Gilbert-Elliot model. We compare the performance of a Kalman filter with and without using MD codes. The improved bounds of expected value of error covariance of the Kalman filter over a large set of packet loss scenarios are given and verified by simulation results.

## 5.1 Formulation and Assumptions for Packet-Based State Estimation

A diagram of the open-loop packet-based state estimation problem is shown in Figure 5.1. The data link from observer to state estimator is not modelled as a single, exclusive communication channel, but rather as a possible path through a large, complex communication network shared with many other users. Also the channel encoder and channel decoder are omitted.

We assume that the dynamic system in Figure 5.1 is presented by a discrete-time linear dynamic system:

$$\begin{aligned} x_{k+1} &= Ax_k + w_k \\ y_k &= Cx_k + v_k \end{aligned} \tag{5.1}$$

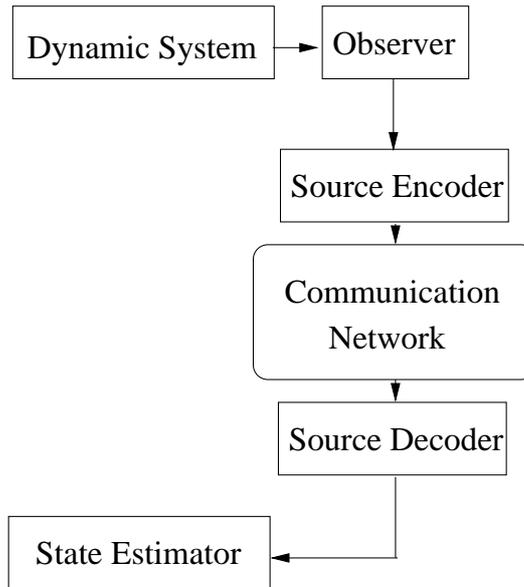


Figure 5.1: Diagram of packet-based state estimation

where  $x_k \in \mathcal{R}^n$  is the state vector,  $y_k \in \mathcal{R}^m$  is the output vector,  $w_k$  and  $v_k$  are Gaussian white noise vectors with zero mean, and covariance matrices are  $Q \geq 0$  and  $G > 0$ , respectively. Suppose that  $A$  is unstable. A standard discrete-time Kalman filter is used as the estimator. It is well known that if the pair  $(A, Q^{\frac{1}{2}})$  is controllable, the pair  $(A, C)$  is detectable, and no measurement is lost, the estimation error covariance of the Kalman filter converges to a unique value from any initial condition.

For the system in Figure 5.1, there are several assumptions for the communication network we need in order to simplify the problem:

- We ignore channel coding and assume that the data packet is either received and decoded successfully or totally lost.
- The estimator does not have a memory of the previous packets it has received. Only the new, “real-time” data is used by the state estimator in each update cycle. If a packet arrives too late (after the update cycle), it is discarded and treated as a dropped packet. The dropped packets will not be retransmitted.
- The network does not provide preferential treatment to any packet. In other words, the network treats each single packet equally without inspecting its content. Thus a

multiple resolution code or a layered source code is not a good choice for us since they mark packets with different priorities according to the contents.

- There is no feedback link from estimator to observer.
- We assume that each data packet is relatively large and the network is running at a high bit rate scenario.

As a quick reminder, we assume that packets are dropped randomly according following models:

- **Bernoulli Packet-Dropping Model.** A Bernoulli random variable  $\gamma_k$  indicates whether the packet  $k$  is received correctly. If it goes through the network successfully, then  $\gamma_k = 1$ , otherwise,  $\gamma_k = 0$ . The sequence  $\gamma_k$  is i.i.d. with probability distribution  $P(\gamma_k = 1) = \lambda$  and  $P(\gamma_k = 0) = (1 - \lambda)$ .
- **Two-state Markov Chain Model.** This model considers the network as a discrete-time Markov chain with two possible states: “good” and “bad.” In the “good” state, the packet is received correctly, and in the “bad” state, the packet is dropped. The network jumps between these two states with transition probability matrix  $Q$  as

$$Q = \begin{bmatrix} q_{00} & q_{01} \\ q_{10} & q_{11} \end{bmatrix} \quad (5.2)$$

where 1 is the good state, 0 is the bad state, and  $q_{ij}$  is the probability from the previous state  $j$  to the next state  $i$ .

## 5.2 Multiple Description Source Codes

The traditional source encoder for the dynamic system in Figure 5.1 is actually a fixed-rate uniform scalar quantizer with  $N$  reconstruction levels. We assume that the initial state of the dynamic system is uniformly distributed on a state space with length  $L$ . The optimal

rate-distortion function for single description source codes is

$$D(R) \geq \frac{L^2}{12} \cdot 2^{-2R} \quad (5.3)$$

where  $R = \log_2(N)$  is the bits per source sample (bpss). For other state distributions, we have similar rate-distortion functions that all decay at speed  $2^{-2R}$ .

MD source codes are designed to achieve good rate-distortion performance over lossy links. The unique feature of MD codes is that instead of using one single description to represent one source sample, MD codes use two or more descriptions. So at the end of the data link, the decoder has much less chance of losing all descriptions. There are two simplest cases to generate an MD code:

- For each source sample, all descriptions are identical to the traditional source coding description. Thus, the decoder can rebuild the source sample even if only one description is received. This case is actually equivalent to sending the same packet multiple times. The cost of the high reliability is the high redundancy and the high demand of the channel bandwidth.
- Divide a traditional source coding description into couple of non-overlapping increments. Each increment is treated as an independent description. This method alleviates the demand of channel bandwidth, but the average distortion at the decoder is greatly increased even if only one description is lost.

Designing useful MD codes requires achieving a balance between the efficiency of the bandwidth and the robustness to the packet loss. In order to keep the total bpss small, the redundancy between the descriptions should be as little as possible. On the other hand, the distortion of the decoder depends on how many descriptions it receives. Normally, the more descriptions the decoder receives for one source sample, the more accurate the output should be. Thus, the fundamental tradeoff in MD coding is finding descriptions that are individually good and sufficiently different at the same time. Also MD codes need to be nonhierarchical so that the receiving order of descriptions is not important.

### 5.2.1 Theoretical Limits of Multiple Description Codes

In this subsection, we introduce some theoretical limits about the bps and average distortion for two-description MD codes. Suppose the input of the MD encoder is a sequence of source sample values  $\{X_k\}$ . The descriptions are  $\{i_k, j_k\}$  and the number of bits for descriptions are  $R_i$  and  $R_j$ , respectively. There exist three cases according to which descriptions are received:

- At step  $k$ , the decoder receives none of the descriptions. We call this the “broken link” case and will discuss it in Section 5.3.
- At step  $k$ , the decoder receives both  $\{i_k\}$  and  $\{j_k\}$ . We call it the “central decoder” case and the average distortion is denoted by  $D_c$ .
- At step  $k$ , the decoder only receives either  $\{i_k\}$  or  $\{j_k\}$ . We call it the “side decoder” case. The average distortions are denoted by  $D_i$  and  $D_j$ , respectively.

The main theoretical problem of MD coding is to determine the boundary of the achievable quintuple  $(R_i, R_j, D_c, D_i, D_j)$ . If  $R_i = R_j$  and  $D_i \approx D_j$ , the two-description MD code is called balanced. According to [112], the achievable rate-distortion region of a two-description MD code for a memoryless unit variance Gaussian source is

$$\begin{cases} D_i \geq 2^{-2R_i} \\ D_j \geq 2^{-2R_j} \\ D_c \geq 2^{-2(R_i+R_j)} \cdot \gamma(D_i, D_j, R_i, R_j) \end{cases} \quad (5.4)$$

where

$$\gamma = \frac{1}{1 - \left( \sqrt{(1-D_i)(1-D_j)} - \sqrt{D_i D_j - 2^{-2(R_i+R_j)}} \right)^2}$$

for  $D_i + D_j < 1 + 2^{-2(R_i+R_j)}$  and  $\gamma = 1$  otherwise.

For packet-based state estimation, balanced two-description MD codes are employed. Assume that  $R_i = R_j = R \gg 1$  and  $D_i = D_j = 2^{-2R(1-\alpha)} \ll 1$  with  $0 < \alpha < 1$ , we have

$$\begin{aligned} \frac{1}{\gamma} &= 1 - \left( (1 - D_i) - \sqrt{D_i^2 - 2^{-4R}} \right)^2 \\ &\approx 1 - ((1 - D_i) - D_i)^2 \approx 4D_i. \end{aligned}$$

Thus,

$$D_c \cdot D_i \gtrsim \frac{1}{4} 2^{-4R}. \quad (5.5)$$

The inequality (5.5) shows a tradeoff between central and side distortions. Compared with inequality (5.3), it is clear that the penalty in the exponential rate of decay of  $D_i$  is exactly the increase in the rate of decay of  $D_c$ . A complete discussion of the rate-distortion region for other types of description codes is outside the scope of this work.

## 5.2.2 Multiple Description Scalar Quantizer

The multiple description scalar quantizer (MDSQ) is used to generate two descriptions for each source sample. This method was proposed and popularized by [113]. Figure 5.2 shows the diagram of a two-description MDSQ:

- Step one: Select a uniform quantizer with an appropriate number of step levels  $N$ . A source sample  $Y$  is quantized by rounding off to the nearest multiple of a step size  $\Delta$  and the index output is  $n$  that satisfies  $0 < n \leq N$ .
- Step two: A pair of indices  $(i, j)$  is assigned to the index  $n$  by using an index mapping matrix. Those two indices are the corresponding descriptions for the source sample  $Y$ .
- Step three: The descriptions  $i$  and  $j$  are encapsulated into two separate data packets.

The index mapping problem in step two is the main part of MDSQ. A  $\sqrt{M} \times \sqrt{M}$  matrix is constructed where  $M \geq N$ . We arrange all the numbers from 1 to  $N$  into the cells in the matrix. Each cell holds one number at most, so each index  $n$  gets a pair of matrix indices  $(i, j)$  according to its location where  $i$  is the row number and  $j$  is the column

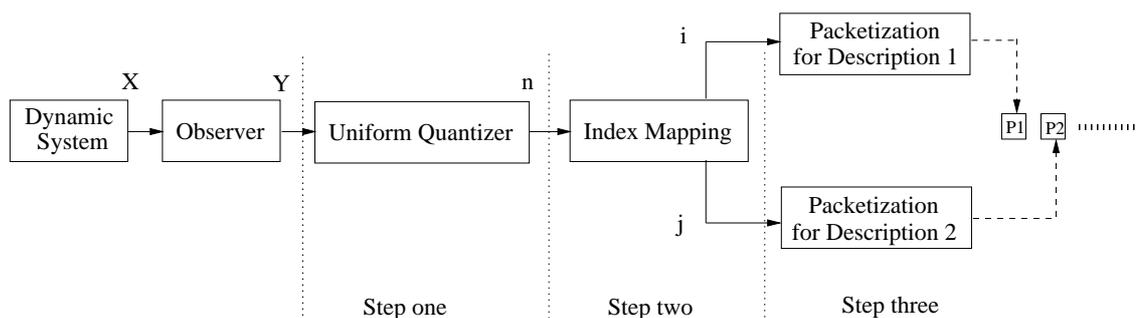


Figure 5.2: Diagram of two-description MD source encoder

number. Using this index mapping matrix, step two transfers each single description  $n$  into two descriptions  $i$  and  $j$ . For a properly designed index mapping matrix (refer to Appendix A for more details), we get

$$\begin{cases} D_c \approx C_0 2^{-2R(1+\alpha)} \\ D_i \approx D_j \approx C_1 2^{-2R(1-\alpha)} \end{cases} \quad (5.6)$$

where  $C_0$  and  $C_1$  are constants. Parameter  $\alpha \in [0, 1]$  is a pre-defined parameter that indicates the tradeoff between the decay speeds of  $D_c$  and  $D_i$ . It is clear that the average distortion of the “central decoder” equals the distortion of the uniform quantizer in step one.

The index mapping method can be extended to other MD codes. Table 5.1 lists the values of the average distortions and bps for some MD codes defined on  $[-5, 5]$ . The first part of Table 5.1 shows examples of average distortions for different description loss cases when we keep  $D_c$  constant, in which a larger bps is needed when the number of descriptions increases. The second part shows that, if we keep bps constant, the distortion increases when the number of descriptions increases. In the table, “lost  $k$ ” means  $k$  descriptions have been lost and “N/A” means not available. Apparently, balanced MD codes can provide various distortion levels corresponding to how many descriptions the decoder receives.

Table 5.1: Average distortion for different MD codes

Coding type	No loss	Lost 1	Lost 2	Lost 3	Total bpss
single description	$8.33 \times 10^{-6}$	N/A	N/A	N/A	10
2-description	$8.33 \times 10^{-6}$	1.56	N/A	N/A	12
3-description	$8.33 \times 10^{-6}$	$4.41 \times 10^{-3}$	1.53	N/A	15
4-description	$8.33 \times 10^{-6}$	$7.46 \times 10^{-3}$	$1.34 \times 10^{-2}$	2.61	20

Coding type	No loss	Lost 1	Lost 2	Lost 3	Total bpss
single description	$4.97 \times 10^{-7}$	N/A	N/A	N/A	12
2-description	$8.33 \times 10^{-6}$	1.56	N/A	N/A	12
3-description	$9.87 \times 10^{-5}$	$1.97 \times 10^{-2}$	2.15	N/A	12
4-description	$9.32 \times 10^{-4}$	$8.04 \times 10^{-2}$	0.113	2.18	12

### 5.2.3 Quantization Noise of MD codes

As discussed in [114], quantization noise of a uniform scalar quantizer with assumptions of small partition cells, reproduction values at cell's midpoints, and large support region can be approximately modelled as an additive uncorrelated white noise to the quantizer input, which is presented in Figure 5.3.

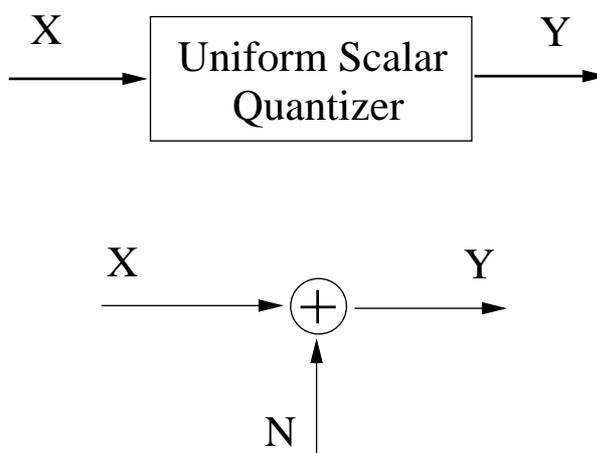


Figure 5.3: Additive noise model of uniform scalar quantization

For balanced MD codes, the central decoder case actually is a uniform scalar quantizer with midpoint outputs and the average distortion is  $D_c \approx \Delta^2/12$ . For the side decoder case, index mapping introduces a slight asymmetry between two side distortions. However, for large bpss, this asymmetry asymptotically disappears. According to previous analysis, we

have

$$D_i \approx D_j \approx C_1 \cdot \left(\frac{1}{12C_0}\right)^{\frac{1-\alpha}{1+\alpha}} \cdot (\Delta^{\frac{1-\alpha}{1+\alpha}})^2.$$

For a balanced two-description MD code,  $\alpha$  is a constant and  $D_i$  will be asymptotically negligible relative to  $(\Delta^{\frac{1-\alpha}{1+\alpha}})^2$ . So as long as the bps  $R_i (= R_j)$  is big enough, the additive noise model is still a good approximation to represent the quantization noise in the side decoder case. From now on, we model the MD quantization noise as Gaussian white noise with zero mean and covariance  $D_c$  for the central decoder case and  $D_i$  for the side decoder case.

## 5.3 Kalman Filtering Utilizing MD Codes

### 5.3.1 Kalman Filtering with i.i.d. Packet Drops

We consider the discrete-time linear dynamical system described by Equation (5.1) and assume that packet drops are described by an i.i.d. Bernoulli random process. Quantitative analysis of the convergence behavior of Kalman filtering was reported in [94]. In this section, we extend this result to the case with MD codes. Two-description balanced MD codes are used so that each  $y_k^l$  in the measurement output  $Y_k = [y_k^1, \dots, y_k^m]$  is encoded by two descriptions  $\{i_k^l, j_k^l\}$ . We organize these descriptions into 2 description vectors as  $\{I_k, J_k\}$  and put them into two different packets. The independent random variables  $\gamma_{I,k}$  and  $\gamma_{J,k}$  are used to indicate whether the description vectors  $I_k$  and  $J_k$  are received correctly. If  $I_k$  is received correctly, then  $\gamma_{I,k} = 1$ , otherwise,  $\gamma_{I,k} = 0$ , and similarly for  $\gamma_{J,k}$ . We assume that both  $\{\gamma_{I,k}\}$  and  $\{\gamma_{J,k}\}$  are i.i.d. Bernoulli random variable sequences with the same probability distribution  $P(\gamma_{I,k} = 1) = P(\gamma_{J,k} = 1) = \lambda$ .

Since  $\gamma_{I,k}$  and  $\gamma_{J,k}$  are independent, we have three measurement rebuilding scenarios. First, we may receive both the descriptions correctly. In this case, the measurement noise is the white noise  $v_k$  plus the central distortion noise. We use  $G_0 = G + D_c$  to indicate the covariance where  $G$  is the observation noise covariance defined in (5.1) and  $D_c$  is the central distortion covariance. Second, we may receive only one description correctly and the measurement noise is  $G_1 = G + D_i$  where  $D_i$  is the side distortion covariance. Third, we

may receive none of the descriptions correctly. In this case, we assume the measurement is corrupted by an infinitely large noise. This corresponds to the “broken link” case in Section 5.2. At step  $k$ , covariance  $G(k)$  for the noise after the decoder is

$$G(k) = \begin{cases} G_0 & \text{with probability } \lambda^2 \\ G_1 & \text{with probability } 2(1 - \lambda)\lambda \\ \sigma^2 I & \text{with probability } (1 - \lambda)^2 \end{cases} \quad (5.7)$$

where  $\sigma \rightarrow \infty$ .

Let  $\gamma_k = [\gamma_{I,k}, \gamma_{J,k}]^T$ ,  $\gamma_0^k = \{\gamma_0, \gamma_1, \dots, \gamma_k\}$ , and  $y_0^k = \{y_0, y_1, \dots, y_k\}$ . We define the following variables:

$$\begin{aligned} \hat{x}_{k|k} &\triangleq E[x_k | y_0^k, \gamma_0^k], \\ P_{k|k} &\triangleq E[(x_k - \hat{x}_{k|k})(x_k - \hat{x}_{k|k})^T | y_0^k, \gamma_0^k], \\ \hat{x}_{k+1|k} &\triangleq E[x_{k+1} | y_0^k, \gamma_0^k], \\ P_{k+1|k} &\triangleq E[(x_{k+1} - \hat{x}_{k+1|k})(x_{k+1} - \hat{x}_{k+1|k})^T | y_0^k, \gamma_0^k]. \end{aligned}$$

In the time update of the Kalman filter, we have

$$\begin{aligned} \hat{x}_{k+1|k} &= A\hat{x}_{k|k}, \\ P_{k+1|k} &= AP_{k|k}A^T + Q, \end{aligned}$$

which are independent of observation data and transmission status at time  $k + 1$ .

The measurement update of the Kalman filter becomes stochastic and depends on the random variables  $\gamma_{I,k}$  and  $\gamma_{J,k}$ . When  $\gamma_{I,k} = 1$  and  $\gamma_{J,k} = 1$ , it is true that

$$\begin{aligned} \hat{x}_{k+1|k+1} &= \hat{x}_{k+1|k} + P_{k+1|k}C^T(CP_{k+1|k}C^T + G_0)^{-1}(y_{k+1} - C\hat{x}_{k+1|k}), \\ P_{k+1|k+1} &= P_{k+1|k} - P_{k+1|k}C^T(CP_{k+1|k}C^T + G_0)^{-1}CP_{k+1|k}. \end{aligned}$$

When  $\gamma_{I,k} = 1$  and  $\gamma_{J,k} = 0$ , or  $\gamma_{I,k} = 0$  and  $\gamma_{J,k} = 1$ , only one description is received

and the covariance of measurement noise is  $G_1$ . The corresponding measurement update is

$$\begin{aligned}\hat{x}_{k+1|k+1} &= \hat{x}_{k+1|k} + P_{k+1|k}C^T(CP_{k+1|k}C^T + G_1)^{-1}(y_{k+1} - C\hat{x}_{k+1|k}), \\ P_{k+1|k+1} &= P_{k+1|k} - P_{k+1|k}C^T(CP_{k+1|k}C^T + G_1)^{-1}CP_{k+1|k}.\end{aligned}$$

When  $\gamma_{I,k} = 0$  and  $\gamma_{J,k} = 0$ , there is no measurement update since no descriptions are received. Thus,

$$\begin{aligned}\hat{x}_{k+1|k+1} &= \hat{x}_{k+1|k}, \\ P_{k+1|k+1} &= P_{k+1|k}.\end{aligned}$$

Combining these three cases, the measurement update process can be rewritten as

$$\begin{aligned}\hat{x}_{k+1|k+1} &= \hat{x}_{k+1|k} + \gamma_{I,k}\gamma_{J,k}P_{k+1|k}C^T(CP_{k+1|k}C^T + G_0)^{-1}(y_{k+1} - C\hat{x}_{k+1|k}) \\ &\quad + (1 - \gamma_{I,k})\gamma_{J,k}P_{k+1|k}C^T(CP_{k+1|k}C^T + G_1)^{-1}(y_{k+1} - C\hat{x}_{k+1|k}) \\ &\quad + \gamma_{I,k}(1 - \gamma_{J,k})P_{k+1|k}C^T(CP_{k+1|k}C^T + G_1)^{-1}(y_{k+1} - C\hat{x}_{k+1|k}), \\ P_{k+1|k+1} &= P_{k+1|k} - \gamma_{I,k}\gamma_{J,k}P_{k+1|k}C^T(CP_{k+1|k}C^T + G_0)^{-1}CP_{k+1|k} \\ &\quad - (1 - \gamma_{I,k})\gamma_{J,k}P_{k+1|k}C^T(CP_{k+1|k}C^T + G_1)^{-1}CP_{k+1|k} \\ &\quad - \gamma_{I,k}(1 - \gamma_{J,k})P_{k+1|k}C^T(CP_{k+1|k}C^T + G_1)^{-1}CP_{k+1|k}.\end{aligned}$$

We simplify the notation and let  $P_k = P_{k|k-1}$ . The Kalman filter recursion thus becomes stochastic and the error covariance evolves as

$$\begin{aligned}P_{k+1} &= AP_kA^T + Q \\ &\quad - \gamma_{I,k}\gamma_{J,k}AP_kC^T[CP_kC^T + G_0]^{-1}CP_kA^T \\ &\quad - (1 - \gamma_{I,k})\gamma_{J,k}AP_kC^T[CP_kC^T + G_1]^{-1}CP_kA^T \\ &\quad - \gamma_{I,k}(1 - \gamma_{J,k})AP_kC^T[CP_kC^T + G_1]^{-1}CP_kA^T.\end{aligned}\tag{5.8}$$

Thus, the sequence of the error covariance  $P_{k=0}^\infty$  is a random process for any given initial value. In order to study the expected value  $E[P_k]$ , we define the modified algebraic Riccati

equation (MARE) for Kalman filters using balanced two-description MD codes as

$$\begin{aligned}
g_\lambda(X) &= AXA^T + Q \\
&\quad -\lambda^2 AX C^T (CXC^T + G_0)^{-1} CX A^T \\
&\quad -2(1-\lambda)\lambda AX C^T (CXC^T + G_1)^{-1} CX A^T
\end{aligned} \tag{5.9}$$

and the expected value of error covariance matrix  $E[P_k]$  evolves according to this MARE.

The following lemmas are used to conduct the theorems in this chapter, and their proofs can be found in Appendix B.

**Lemma 5.3.1.** *Consider operator*

$$\begin{aligned}
\phi(K_0, K_1, X) &= (1-\lambda)^2 (AXA^T + Q) + \lambda^2 (F_0 X F_0^T + V_0) \\
&\quad + 2(1-\lambda)\lambda (F_1 X F_1^T + V_1)
\end{aligned}$$

where  $F_0 = A + K_0 C$ ,  $F_1 = A + K_1 C$ ,  $V_0 = Q + K_0 G_0 K_0^T$ , and  $V_1 = Q + K_1 G_1 K_1^T$ . Assume  $X \in \{S \in \mathcal{R}^{n \times n} | S \geq 0\}$ ,  $G_0 > 0$ ,  $G_1 > 0$ ,  $Q > 0$ , and  $(A, Q^{\frac{1}{2}})$  is controllable. Then the following facts are true:

- (a) With  $K_{x0} = -AXC^T(CXC^T + G_0)^{-1}$  and  $K_{x1} = -AXC^T(CXC^T + G_1)^{-1}$ ,  $g_\lambda(X) = \phi(K_{x0}, K_{x1}, X)$ ;
- (b)  $g_\lambda(X) = \min_{(K_0, K_1)} \phi(K_0, K_1, X) \leq \phi(K_0, K_1, X)$  for any  $(K_0, K_1)$ ;
- (c) If  $X \leq Y$ , then  $g_\lambda(X) \leq g_\lambda(Y)$ ;
- (d) If  $\lambda_1 \leq \lambda_2$ , then  $g_{\lambda_1}(X) \geq g_{\lambda_2}(X)$ ;
- (e) If  $\alpha \in [0, 1]$ , then  $g_\lambda(\alpha X + (1-\alpha)Y) \geq \alpha g_\lambda(X) + (1-\alpha)g_\lambda(Y)$ ;
- (f)  $g_\lambda(X) \geq (1-\lambda)^2 AXA^T + Q$ ;
- (g) If  $\bar{X} \geq g_\lambda(\bar{X})$ , then  $\bar{X} \geq 0$ ;
- (h) If  $X$  is a random variable, then  $(1-\lambda)^2 AE[X]A^T + Q \leq E[g_\lambda(X)] \leq g_\lambda(E[X])$ .

**Lemma 5.3.2.** Let  $X_{k+1} = h(X_k)$  and  $Y_{k+1} = h(Y_k)$ . If  $h(X)$  is a monotonically increasing function, then:

$$X_1 \geq X_0 \Rightarrow X_{k+1} \geq X_k, \forall k \geq 0;$$

$$X_1 \leq X_0 \Rightarrow X_{k+1} \leq X_k, \forall k \geq 0;$$

$$X_0 \leq Y_0 \Rightarrow X_k \leq Y_k, \forall k \geq 0.$$

**Lemma 5.3.3.** Define the linear operator  $L(Y) = (1-\lambda)^2 AYA^T + \lambda^2 F_0 Y F_0^T + 2(1-\lambda)\lambda F_1 Y F_1^T$  and suppose there exists  $\bar{Y} > 0$  such that  $\bar{Y} > L(\bar{Y})$ .

(a) For all  $W \geq 0$ ,  $\lim_{k \rightarrow \infty} L^k(W) = 0$ ;

(b) Let  $V \geq 0$  and consider the linear system  $Y_{k+1} = L(Y_k) + V$  initial at  $Y_0$ , then the sequence  $\{Y_k\}$  is bounded.

**Lemma 5.3.4.** Suppose there exists  $\bar{K}_0, \bar{K}_1$ , and  $\bar{P} > 0$  such that

$$\bar{P} > \phi(\bar{K}_0, \bar{K}_1, \bar{P}),$$

then for any initial value  $P_0$ , the sequence  $P_k = g_\lambda^k(P_0)$  is bounded, i.e., there exists  $M_{P_0} \geq 0$  dependent of  $P_0$  such that

$$P_k \leq M_{P_0}, \forall k.$$

The first theorem listed below states the uniqueness of the MARE solution if it exists.

**Theorem 5.3.1.** Consider the operator

$$\begin{aligned} \phi(K_0, K_1, X) = & (1-\lambda)^2 (AXA^T + Q) \\ & + \lambda^2 (F_0 X F_0^T + V_0) \\ & + 2(1-\lambda)\lambda (F_1 X F_1^T + V_1) \end{aligned} \quad (5.10)$$

where  $F_0 = A + K_0 C$ ,  $F_1 = A + K_1 C$ ,  $V_0 = Q + K_0 G_0 K_0^T$ , and  $V_1 = Q + K_1 G_1 K_1^T$ . Suppose there exist  $K_0, K_1$ , and  $P > 0$  such that  $P > \phi(K_0, K_1, P)$ . Then, for any initial condition  $P_0 \geq 0$ , the iteration  $P_{k+1} = g_\lambda(P_k)$  converges to the unique positive semi-definite solution  $\bar{P}$  of MARE (5.9), i.e.,

$$\lim_{k \rightarrow \infty} P_k = \lim_{k \rightarrow \infty} g_\lambda^k(P_0) = \bar{P} \geq 0$$

where  $\bar{P} = g_\lambda(\bar{P})$ .

**Proof** First, we show that the MARE converges with initial value  $Q_0 = 0$ . Let  $Q_k = g_\lambda(Q_{k-1}) = g_\lambda^k(Q_0)$ , then  $Q_1 \geq Q_0 = 0$  and

$$Q_1 = g_\lambda(Q_0) \leq g_\lambda(Q_1) = Q_2.$$

By induction, we know that the sequence  $\{Q_k\}$  is nondecreasing. Also by Lemma 5.3.4,  $\{Q_k\}$  is bounded and there exists an  $M_{Q_0}$  such that  $Q_k \leq M_{Q_0}$  for any  $k$ . Therefore, the sequence converges and

$$\lim_{k \rightarrow \infty} Q_k = \bar{P} \geq 0$$

where  $\bar{P}$  is a fixed point of the iteration  $\bar{P} = g_\lambda(\bar{P})$ .

Next we show that the iteration  $\bar{G}_k = g_\lambda^k(\bar{G}_0)$  initialized at  $G_0 \geq \bar{P}$  also converges to  $\bar{P}$ . Since  $G_1 = g_\lambda(G_0) \geq g_\lambda(\bar{P}) = \bar{P}$ ,  $G_k \geq \bar{P}$  for any  $k$ . Also

$$\begin{aligned} 0 \leq G_{k+1} - \bar{P} &= g_\lambda(G_k) - g_\lambda(\bar{P}) \\ &= \phi(K_{G_k}, K_{G_{k+1}}, G_k) - \phi(K_{\bar{P}_0}, K_{\bar{P}_1}, \bar{P}) \\ &\leq \phi(K_{\bar{P}_0}, K_{\bar{P}_1}, G_k) - \phi(K_{\bar{P}_0}, K_{\bar{P}_1}, \bar{P}) \\ &= \hat{L}(G_k - \bar{P}) \end{aligned}$$

where  $\hat{L}$  has a similar form as the operator  $L$  in Lemma 5.3.3. Note that

$$\bar{P} = g_\lambda(\bar{P}) > \hat{L}(\bar{P}),$$

thus,  $\hat{L}$  meets all the conditions in Lemma 5.3.3. Using the same argument, we have, for any  $Y \geq 0$ ,

$$\lim_{k \rightarrow \infty} \hat{L}^k(Y) = 0.$$

So we get  $0 \leq \lim_{k \rightarrow \infty} (G_k - \bar{P}) = 0$ , i.e., the sequence  $G_k$  converges to  $\bar{P}$ .

As the last part, we show that, for any initial condition  $P_0 \geq 0$ , the iteration  $P_k = g_\lambda^k(P_0)$  converges to  $\bar{P}$ . Let  $G_0 = P_0 + \bar{P} \geq \bar{P}$ , then  $0 \leq Q_0 \leq P_0 \leq G_0$ , by induction, we have  $0 \leq Q_k \leq P_k \leq G_k$ . Since  $\{Q_k\}$  and  $\{G_k\}$  converges to  $\bar{P}$ ,  $\{P_k\}$  also converges to  $\bar{P}$  and the

result follows. □

The second theorem states the conditions for MARE convergence.

**Theorem 5.3.2.** *If  $(A, Q^{\frac{1}{2}})$  is controllable,  $(A, C)$  is detectable, and  $A$  is unstable, then there exists a  $\lambda_c \in [0, 1)$  such that*

(a) *For  $0 \leq \lambda \leq \lambda_c$ , there exists some initial condition  $P_0 \geq 0$  such that  $E[P_k]$  diverges when  $k \rightarrow +\infty$ , i.e., there does not exist a matrix  $M_{P_0}$  such that  $E[P_k] \leq M_{P_0}$  for any  $k > 0$ ;*

(b) *For  $\lambda_c < \lambda \leq 1$ ,  $E[P_k] \leq M_{P_0}$  for any  $k > 0$  and any initial condition  $P_0 \geq 0$ ;*

where  $M_{P_0} > 0$  depends on the initial condition  $P_0$ .

**Proof** Obviously there are two special cases:

- When  $\lambda = 1$ , the MARE reduces to the standard Algebraic Riccati Equation (ARE) and it converges to a unique positive semi-definite solution.
- When  $\lambda = 0$ , all the packets are lost. Since  $A$  is unstable, the covariance matrix diverges for some initial values.

Next, we need to show that there exists a single point of transition between the two cases. Suppose for  $0 < \lambda_1 \leq 1$ ,  $E_{\lambda_1}[P_k]$  is bounded for any initial values. Then for any  $\lambda_2 > \lambda_1$ , we have

$$E_{\lambda_1}[P_k] = E[g_{\lambda_1}(P_k)] \geq E[g_{\lambda_2}(P_k)] = E_{\lambda_2}[P_k].$$

So  $E_{\lambda_2}[P_k]$  is also bounded. Now we can choose

$$\lambda_c = \{\inf \lambda^* : \lambda > \lambda^* \Rightarrow E_{\lambda}[P_k] \text{ is bounded for any initial value } P_0 \geq 0\}$$

and finish the proof. □

This theorem claims that there exists a critical value  $\lambda_c$  of the packet receiving probability. If  $\lambda$  is smaller than  $\lambda_c$ , MARE (5.9) does not converge and the expected value of error covariance matrix will diverge.

**Theorem 5.3.3.** *Let*

$$\begin{aligned}\underline{\lambda} &= \arg \inf_{\lambda} [\exists \hat{S} \geq 0 \mid \hat{S} = (1 - \lambda)^2 A \hat{S} A^T + Q] = 1 - \frac{1}{a} \\ \bar{\lambda} &= \arg \inf_{\lambda} [\exists \hat{X} \geq 0 \mid \hat{X} > g_{\lambda}(\hat{X})] \\ &= \arg \inf_{\lambda} [\exists (\hat{K}_0, \hat{K}_1, \hat{X} \geq 0) \mid \hat{X} > \phi(\hat{K}_0, \hat{K}_1, \hat{X})]\end{aligned}$$

where  $a = \rho(A)$  is the spectral radius of  $A$ . Then

$$\underline{\lambda} \leq \lambda_c \leq \bar{\lambda}. \quad (5.11)$$

**Proof** For the lower bound of  $\lambda_c$ , we define the Lyapunov operator  $\mathcal{M}(X) = \bar{A}X\bar{A}^T + Q$  where  $\bar{A} = (1 - \lambda)A$ . If  $(A, Q^{\frac{1}{2}})$  is controllable,  $(\bar{A}, Q^{\frac{1}{2}})$  is also controllable. Then the  $\hat{S} = \mathcal{M}(\hat{S})$  has a unique strictly positive definite solution  $\hat{S}$  if and only if  $\max_i |\sigma_i(\bar{A})| < 1$ , so we get  $\underline{\lambda} = 1 - \frac{1}{a}$ . Consider the iteration  $S_{t+1} = \mathcal{M}(S_t)$  for any  $\lambda > \underline{\lambda}$ , it converges. While for  $\lambda \leq \underline{\lambda}$ , it is unstable and  $S_k$  tends to infinity for any initial values.

For the mean value of the error covariance matrix  $E[P_k]$  initialized at  $E[P_0] \geq 0$ , consider  $0 = S_0 \leq E[P_0]$ , it's easy to show that

$$\begin{aligned}S_k \leq E[P_k] \Rightarrow S_{k+1} &= \mathcal{M}(S_k) \\ &\leq (1 - \lambda)^2 A E[P_k] A^T + Q \\ &\leq E[g_{\lambda}(P_k)] = E[P_{k+1}].\end{aligned}$$

By induction, it is obvious that when  $\lambda < \underline{\lambda}$ ,  $\lim_{k \rightarrow \infty} E[P_k] \geq \lim_{k \rightarrow \infty} S_k = \infty$ . This implies that for any initial condition  $E[P_k]$  is unbounded for  $\lambda < \underline{\lambda}$ , therefore  $\underline{\lambda} \leq \lambda_c$ .

For the upper bound of  $\lambda_c$ , consider the sequence  $V_{k+1} = g_{\lambda}(V_k)$  and  $V_0 = E[P_0] \geq 0$ , we have

$$\begin{aligned}E[P_k] \leq V_k \Rightarrow E[P_{k+1}] &= E[g_{\lambda}(P_k)] \\ &\leq g_{\lambda}(E[P_k]) \\ &\leq g_{\lambda}(V_k) = V_{k+1}.\end{aligned}$$

A simple induction shows that for any  $k$ ,  $V_k \geq E[P_k]$ . So for  $\lambda > \bar{\lambda}$ , according to Lemma 5.3.1 part (g), there exists  $\bar{X} > 0$ . Therefore all conditions of Lemma 5.3.4 are satisfied and

we have

$$E[P_k] \leq V_k \leq M_{V_0}$$

for any  $k$ . This shows that  $\lambda_c \leq \bar{\lambda}$ . □

This theorem states the upper and lower bounds for  $\lambda_c$ . The lower bound is in a closed form. According to the next theorem, we can reformulate the computation of  $\bar{\lambda}$  as an linear matrix inequalities (LMI) feasible problem.

**Theorem 5.3.4.** *Assume that  $(A, Q^{\frac{1}{2}})$  is controllable and  $(A, C)$  is detectable, then the following statements are equivalent:*

- (a)  $\exists \bar{X} > 0$  such that  $\bar{X} > g_\lambda(\bar{X})$ ;
- (b)  $\exists \bar{K}_0, \bar{K}_1$ , and  $\bar{X} > 0$  such that  $\bar{X} > \phi(\bar{K}_0, \bar{K}_1, \bar{X})$ ;
- (c)  $\exists \bar{Z}_0, \bar{Z}_1$  and  $0 < \bar{Y} \leq I$  such that

$$\Psi_\lambda(\bar{Y}, \bar{Z}_0, \bar{Z}_1) > 0$$

where

$$\Psi_\lambda = \begin{bmatrix} Y & \Delta(Y, Z_1) & \Omega(Y, Z_0) & \Pi(Y) \\ \Delta(Y, Z_1)^T & Y & 0 & 0 \\ \Omega(Y, Z_0)^T & 0 & Y & 0 \\ \Pi(Y)^T & 0 & 0 & Y \end{bmatrix},$$

$$\Delta(Y, Z_1) = \sqrt{2(1-\lambda)\lambda}(YA + Z_1C), \quad \Omega(Y, Z_0) = \lambda(YA + Z_0C), \quad \text{and} \quad \Pi(Y) = (1-\lambda)YA.$$

**Proof** Using Lemma 5.3.1, it is easy to show that if there exists an  $\bar{X} > 0$  such that  $\bar{X} > g_\lambda(\bar{X})$ ,  $\bar{X} > g_\lambda(\bar{X}) = \phi(K_{\bar{X}0}, K_{\bar{X}1}, \bar{X})$ . Also it is obvious that  $\bar{X} > \phi(K_0, K_1, \bar{X}) \geq g_\lambda(\bar{X})$ , so (a) is equivalent to (b). The only trick we need for the remaining proof is to use Schur

complement decomposition to obtain the function  $\Psi_\lambda$ . Please note that

$$\begin{aligned}
\phi(K_0, K_1, X) &= (1 - \lambda)^2(AXA^T + Q) + \lambda^2(F_0XF_0^T + V_0) \\
&\quad + 2(1 - \lambda)\lambda(F_1XF_1^T + V_1) \\
&= (1 - \lambda)^2AXA^T + Q + \lambda^2F_0XF_0^T \\
&\quad + 2(1 - \lambda)\lambda F_1XF_1^T + \lambda^2K_0G_0K_0^T \\
&\quad + 2(1 - \lambda)\lambda K_1G_1K_1^T.
\end{aligned}$$

The part (b) is equivalent to

$$\begin{bmatrix} X - (1 - \lambda)^2AXA^T + \lambda^2F_0XF_0^T & \sqrt{2(1 - \lambda)}\lambda F_1 \\ \sqrt{2(1 - \lambda)}\lambda F_1^T & X^{-1} \end{bmatrix} > 0.$$

Using Schur complement decomposition two more times we obtain

$$\begin{bmatrix} X & \sqrt{2(1 - \lambda)}\lambda F_1 & \lambda F_0 & (1 - \lambda)A \\ \sqrt{2(1 - \lambda)}\lambda F_1^T & X^{-1} & 0 & 0 \\ \lambda F_0^T & 0 & X^{-1} & 0 \\ (1 - \lambda)A^T & 0 & 0 & X^{-1} \end{bmatrix} > 0.$$

Let  $Y = X^{-1}$ ,  $Z_1 = X^{-1}K_1$ , and  $Z_0 = X^{-1}K_0$ , we get

$$\begin{bmatrix} Y & \sqrt{2(1 - \lambda)}\lambda I_1 & \lambda I_0 & (1 - \lambda)YA \\ \sqrt{2(1 - \lambda)}\lambda I_1^T & Y & 0 & 0 \\ \lambda I_0^T & 0 & Y & 0 \\ (1 - \lambda)A^TY & 0 & 0 & Y \end{bmatrix} > 0$$

where  $I_1 = YA + Z_1C$  and  $I_0 = YA + Z_0C$ , and this is what we define as  $\Psi_\lambda(Y, Z_0, Z_1)$ . Since  $\Psi_\lambda(aY, Z_0, Z_1) = a\Psi_\lambda(Y, Z_0, Z_1)$ , so  $Y$  can be restricted to  $0 < Y \leq I$ .  $\square$

When  $C$  is invertible, we choose  $K_0 = K_1 = -AC^{-1}$  to make  $F_0 = F_1 = 0$  and the LMI in Theorem 5.3.4 is equivalent to

$$X - (1 - \lambda)^2AXA^T > 0.$$

Since the solution  $X \geq 0$  exists if and only if  $(1 - \lambda)A$  is stable, i.e., all the magnitudes of eigenvalues of  $(1 - \lambda)A$  are smaller than 1, we obtain  $\bar{\lambda} = \underline{\lambda} = 1 - \frac{1}{a}$ . This value is smaller than the critical value  $1 - \frac{1}{a^2}$  discussed in [94] where single description codes are used. So using MD codes pushes  $\lambda_c$  to a smaller value and guarantees the convergence over a larger packet-dropping scenario.

The following theorem gives the upper and lower bounds on the expected value of error covariance matrix when the MARE converges. The lower bound  $\bar{S}$  can be computed by standard Lyapunov equation solvers and the upper bound  $\bar{V}$  can be either computed via iterating  $V_{t+1} = g_\lambda(V_t)$  from any initial condition or transferred to a semi-definite programming (SDP) problem.

**Theorem 5.3.5.** *Assume  $(A, Q^{\frac{1}{2}})$  is controllable,  $(A, C)$  is detectable, and  $\bar{\lambda} < \lambda$ , then for any initial condition  $E[P_0] \geq 0$ ,*

$$0 \leq \bar{S} \leq \lim_{k \rightarrow \infty} E[P_k] \leq \bar{V}$$

where  $\bar{S}$  and  $\bar{V}$  are solutions of the equations  $\bar{S} = (1 - \lambda)^2 A \bar{S} A^T + Q$  and  $\bar{V} = g_\lambda(\bar{V})$ , respectively.

**Proof** Let  $S_{k+1} = \mathcal{M}(S_k) = (1 - \lambda)^2 A S_k A^T + Q$  and  $V_{k+1} = g_\lambda(V_k)$  with initial conditions  $S_0 = 0$  and  $V_0 = E[P_0] \geq 0$ . By Theorem 5.3.3, we obtain

$$S_k \leq E[P_k] \leq V_k$$

for any  $k$ . According to Theorem 5.3.1,  $\lim_{k \rightarrow \infty} V_k = \bar{V}$  where  $\bar{V} = g_\lambda(\bar{V})$ . Also since  $(A, Q^{\frac{1}{2}})$  is controllable and all the magnitudes of the eigenvalues of  $(1 - \lambda)A$  are smaller than 1, the sequence of the Lyapunov iteration converges to the strictly positive definite solution of the Lyapunov function, i.e.,  $\lim_{k \rightarrow \infty} S_k = \bar{S} > 0$ . Therefore we can conclude that

$$0 < \bar{S} = \lim_{k \rightarrow \infty} S_k \leq \lim_{k \rightarrow \infty} E[P_k] \leq \lim_{k \rightarrow \infty} V_k = \bar{V}. \quad \square$$

### 5.3.2 Discussion for Generalized MARE

The MARE for Kalman filters can be generalized to other i.i.d. random processes. Suppose that the estimator can access observation data according to a random variable  $\gamma_k$  that has the distribution  $\{p_1, \dots, p_m\}$  over the set  $\{1, \dots, m\}$ . The covariance  $G(k)$  for the observation noise is dependent on  $\gamma_k$  as

$$G(k) = \begin{cases} G_1 & \text{with probability } p_1, \\ G_2 & \text{with probability } p_2, \\ \vdots & \\ G_m & \text{with probability } p_m. \end{cases} \quad (5.12)$$

Then, the generalized MARE for the expected error covariance is

$$g(X) = AXA^T + Q - \sum_{i=1}^m p_i \cdot AXC^T(CXC^T + G_i)^{-1}CXA^T. \quad (5.13)$$

For an unstable  $A$ , the term  $AXA^T + Q$  enlarges the value of  $X$  during the iteration. On the other hand, the term  $-\sum_i p_i \cdot AXC^T(CXC^T + G_i)^{-1}CXA^T$  reduces  $X$ . When the two effects reach a balance, we obtain the solution of the generalised MARE (5.13). If all of the covariances  $G_i$  are bounded, the solution of MARE is bounded, i.e., the error covariance of the Kalman filter does not diverge since

$$g(X) \leq AXA^T + Q - AXC^T(CXC^T + \max_i(G_i))^{-1}CXA^T. \quad (5.14)$$

However, if there exists a  $G_i = \infty \cdot I$ , then the solution may diverge with respect to  $p_i$ . From that perspective, MARE (5.9) discussed in previous subsection is a special case with  $G_3 = \infty \cdot I$  and  $p_3 = (1 - \lambda)^2$ .

Without loss of generality, we assume  $G_i$  are bounded except that  $G_m = \infty \cdot I$  in Equation (5.12) and we have the following theorems for the generalized MARE (5.13).

**Theorem 5.3.6.** Consider the operator

$$\begin{aligned} \phi(K_1, \dots, K_{m-1}, X) &= p_m \cdot (AXA^T + Q) \\ &+ \sum_{i=1}^{m-1} p_i \cdot (F_i X F_i^T + V_i) \end{aligned} \quad (5.15)$$

where  $F_i = A + K_i C$  and  $V_i = Q + K_i G_i K_i^T$ . Suppose there exist  $\{K_0, \dots, K_{m-1}\}$ , and  $P > 0$  such that  $P > \phi(K_1, \dots, K_{m-1}, P)$ . Then, for any initial condition  $P_0 \geq 0$ , the iteration  $P_{k+1} = g(P_k)$  converges to the unique positive semi-definite solution  $\bar{P}$  of MARE (5.13), i.e.,

$$\lim_{k \rightarrow \infty} P_k = \lim_{k \rightarrow \infty} g^k(P_0) = \bar{P} \geq 0$$

where  $\bar{P} = g_\lambda(\bar{P})$ .

**Proof** All the results in Lemma 5.3.1, 5.3.3, and 5.3.4 hold for the new operator (5.15). Therefore, it is straightforward to get the result using a similar argument as in Theorem 5.3.1.  $\square$

**Theorem 5.3.7.** If  $(A, Q^{\frac{1}{2}})$  is controllable,  $(A, C)$  is detectable, and  $A$  is unstable, then there exists a  $p_c \in (0, 1)$  such that

- (a) For  $p_m \geq p_c$ , there exists some initial condition  $P_0 \geq 0$  such that  $E[P_k]$  diverges when  $k \rightarrow +\infty$ ;
- (b) For  $p_m < p_c$ ,  $E[P_k]$  is bounded for any  $k > 0$  and any initial condition  $P_0 \geq 0$ .

Moreover, let

$$\begin{aligned} \bar{p} &= \arg \sup_p [\exists \hat{S} \geq 0 \mid \hat{S} = p \cdot A \hat{S} A^T + Q] = \frac{1}{a^2} \\ \underline{p} &= \arg \sup_p [\exists \hat{X} \geq 0 \mid \hat{X} > g(\hat{X})] \end{aligned}$$

where  $a = \rho(A)$  is the spectral radius of  $A$ . Then

$$\underline{p} \leq p_c \leq \bar{p}. \quad (5.16)$$

**Proof** With a similar argument as in Theorem 5.3.3, we define the Lyapunov operator as  $\mathcal{M}(X) = \bar{A} X \bar{A}^T + Q$  where  $\bar{A} = \sqrt{p_m} A$  for the upper bound of  $p_m$ . For the lower bound, we use the sequence  $V_{k+1} = g(V_k)$  and  $V_0 = E[P_0] \geq 0$ .  $\square$

**Theorem 5.3.8.** *Assume  $(A, Q^{\frac{1}{2}})$  is controllable,  $(A, C)$  is detectable, and  $p_m < \underline{p}$ , then for any initial condition  $E[P_0] \geq 0$ ,*

$$0 \leq \bar{S} \leq \lim_{k \rightarrow \infty} E[P_k] \leq \bar{V}$$

where  $\bar{S}$  and  $\bar{V}$  are solutions of the equations  $\bar{S} = p_m \cdot A\bar{S}A^T + Q$  and  $\bar{V} = g(\bar{V})$ , respectively.

**Proof** Let  $S_{k+1} = p_m \cdot AS_kA^T + Q$  and  $V_{k+1} = g(V_k)$ . According to the discussion in previous subsection, we have  $S_k \leq E[P_k] \leq V_k$ . Since  $S_k$  converges to  $\bar{S}$  and  $V_k$  converges to  $\bar{V}$ , it is true that

$$\bar{S} \leq \lim_{k \rightarrow \infty} E[P_k] \leq \bar{V}.$$

□

For a generalized MARE, the convergence of the solution relies on the probability associated with the infinite noise. For the packet-based state estimation, this probability is equivalent to the chance that estimator does not receive any information about the observation. Balanced two-description MD codes actually reduce the chance by sending two small packets for each observation. Also, MD codes efficiently utilize the communication bandwidth. The performance of Kalman filters with other MD codes can be quantitatively analyzed similarly as long as the probability distribution and corresponding noises is determined.

### 5.3.3 Kalman Filtering with Bursty Packet Drops

As mentioned in Chapter 2, bursty packet drops can be modelled as a two-state Markov chain with transition probability matrix  $Q$  given by Equation (5.2). For the case of using balanced two-description MD codes, we are interested in a four-state Markov chain where the states correspond to both packets lost, only the description  $I_k$  is lost, only the description  $J_k$  is lost, and no packet is lost. <sup>1</sup> The transition probability matrix of this chain is given as

---

<sup>1</sup>This is a joint work with Vijay Gupta.

$$Q_{MD} = \begin{bmatrix} q_{00}^2 & q_{00}^2 & q_{01}q_{00} & q_{01}q_{00} \\ q_{01}q_{10} & q_{01}q_{10} & q_{11}q_{01} & q_{11}q_{01} \\ q_{10}q_{00} & q_{10}q_{00} & q_{01}q_{10} & q_{01}q_{10} \\ q_{10}q_{11} & q_{10}q_{11} & q_{11}^2 & q_{11}^2 \end{bmatrix}. \quad (5.17)$$

Note that the state in which both description packets are lost is equivalent to no observation coming through, while the other states correspond to the system being observed.

If the Markov chain is stationary, state probabilities tend to form a stationary distribution as  $k \rightarrow \infty$ . However, we normally cannot study the problem using the generalized MARE over this stationary distribution because the explicit evaluation of the expected value of the ARE is intractable [115]. However, this problem is mathematically equivalent to the random sensor scheduling problem in sensor networks. Consider the system

$$x_{k+1} = Ax_k + w_k$$

being observed through  $n$  sensors with the  $i$ th sensor of the form

$$y_k^i = C^i x_k + v_k^i. \quad (5.18)$$

Suppose only one sensor can be active at any time instant and the choice of the sensor is done according to a Markov chain. We denote the Ricatti update in error covariance by  $f_i(\cdot)$  when the  $i$ th sensor is used and denote

$$f_i^k(\cdot) = \underbrace{f_i(f_i(\dots(\cdot)\dots))}_{k \text{ times}}.$$

The expected error covariance at time step  $k$  is denoted by  $E[P_k]$ . Probability of the network in Markov state  $j$  at time  $k$  is denoted by  $\pi_k^j$  and  $q_{ij}$  is the probability of the network state is  $i$  at time  $k+1$  given the network state is  $j$  at time  $k$ .

**Lemma 5.3.5.** *For any Ricatti update operator  $f_i(P)$ , we have*

$$(a) \quad f_i(P) \geq Q;$$

(b) If  $X < Y$ , then  $f_i(X) \leq f_i(Y)$ ;

(c)  $f_i(P)$  is concave with respect to  $P$ .

With single-description codes, according to Equation (5.18), packet-dropping can be treated as the observation jumps between two sensors that have the same  $C^i$  matrices and different Gaussian noises with covariance  $G_0$  and  $\infty \cdot I$ , respectively. The Kalman filter error covariance updates are

$$\begin{cases} f_0(P) = APA^T + Q \\ f_1(P) = APA^T + Q - APC^T(CPC^T + G_0)^{-1}CPA^T. \end{cases}$$

Similar to the i.i.d. Bernoulli model, we discuss the conditions and the upper/lower bounds for expected values of estimation error covariances converging.

**Theorem 5.3.9.** *When using a single-description code and with the Markov probability transition matrix given by Equation (5.2), the lower bound for  $E[P_k]$  is  $Y_k$  where*

$$\begin{aligned} Y_k = & q_{00}^k \pi_0^0 f_0^k(P_0) + \pi_k^1 f_1(Q) \\ & + \sum_{i=1}^{k-1} q_{00}^i \left( \pi_{k+1-i}^0 - q_{00} \cdot \pi_{k-i}^0 \right) f_0^i(Q). \end{aligned} \quad (5.19)$$

The upper bound is  $X_k$  where

$$X_k = \sum_{j=0}^1 \sum_{i=0}^1 f_j(X_{k-1}^i) q_{ji} \pi_{k-1}^i \quad (5.20)$$

and  $X_{k-1}^i = E[X_{k-1} | \text{state is } i \text{ at time } (k-2)]$ .

**Proof** Suppose  $k$  starts from 1, and for any  $k$ , we define event  $E_i$ , which means that the last packet was received at time  $k-i$  where  $i \in [0, \dots, k]$ . So the probability of  $E_i$  is

$$P_i = \begin{cases} q_{00}^k \pi_0^0 & i = k \\ q_{00}^{i-1} q_{01} \pi_{k-i}^1 & 0 < i < k \\ \pi_k^1 & i = 0 \end{cases}$$

and the error covariance  $P_k$  if  $E_i$  happens is

$$P_k|E_i = \begin{cases} f_0^k(P_0) & i = k \\ f_0^i(f_1(P_{k-i})) & 0 < i < k \\ f_1(P_k) & i = 0. \end{cases}$$

So

$$\begin{aligned} E[P_k] &= \sum_{i=0}^k p_i \cdot P_k|E_i \\ &= q_{00}^k \pi_0^0 f_0^k(P_0) + \sum_{i=1}^{k-1} q_{00}^i q_{01} \pi_{k-i}^1 f_0^i(f_1(P_{k-i})) \\ &\quad + \pi_k^1 f_1(P_k). \end{aligned}$$

According to Lemma 5.3.5,  $f_1(P_{k-1}) \geq Q$ , so

$$\begin{aligned} E[P_k] &\geq q_{00}^k \pi_0^0 f_0^k(P_0) + \pi_k^1 f_1(Q) \\ &\quad + \sum_{i=1}^{k-1} q_{00}^i q_{01} \pi_{k-i}^1 f_0^i(Q) \\ &= q_{00}^k \pi_0^0 f_0^k(P_0) + \pi_k^1 f_1(Q) \\ &\quad + \sum_{i=1}^{k-1} q_{00}^i (\pi_{k+1-i}^0 - q_{00} \cdot \pi_{k-i}^0) f_0^i(Q). \end{aligned}$$

For the upper bound, let us denote  $S_k$  is the network state at time  $k$ . For a single-description code,  $S_k \in [0, 1]$ . Then  $E[P_k] = \sum_{j=0}^1 \pi_k^j \cdot E[P_k|S_k = j]$ . Also

$$\begin{aligned} &\pi_k^j \cdot E[P_k|S_k = j] \\ &= \pi_k^j \sum_{i=0}^1 E[P_k|S_k = j, S_{k-1} = i] \cdot p(S_{k-1} = i|S_k = j) \\ &= \sum_{i=0}^1 E[f_j(P_{k-1})|S_{k-1} = i] q_{ji} \pi_{k-1}^i \\ &\leq \sum_{i=0}^1 f_j([P_{k-1}|S_{k-1} = i]) q_{ji} \pi_{k-1}^i \end{aligned}$$

since  $f_j(\cdot)$  is concave. □

**Proposition 5.3.10.** *A sufficient condition for divergence of expected error covariance is*

$$q_{00} \cdot a^2 > 1, \tag{5.21}$$

where  $a = \rho(A)$ .

Using a balanced two-MD code, the corresponding sensor selection problem has four sensors that have the same  $C$  matrices, and noise covariances are  $G_0$ ,  $G_1$ ,  $G_1$ , and  $\infty \cdot I$ , respectively. The Ricatti updates are

$$\begin{cases} f_0(P) = APA^T + Q \\ f_1(P) = APA^T + Q - APC^T(CPC^T + G_1)^{-1}CPA^T \\ f_2(P) = f_1(P) \\ f_3(P) = APA^T + Q - APC^T(CPC^T + G_0)^{-1}CPA^T. \end{cases}$$

Using the same approach, we get

**Proposition 5.3.11.** *When using a two-description code and with the underlying Markov probability transition matrix given by (5.2), the lower bound for  $E[P_k]$  is  $Y_k$  where*

$$\begin{aligned} Y_k = & q_{00}^{2k} \pi_0^k f_0^k(P_0) + \sum_{j=1}^3 \pi_k^j f_j(Q) \\ & + \sum_{i=1}^{k-1} q_{00}^{2i} (\pi_{k+1-i}^0 - q_{00}^2 \cdot \pi_{k-i}^0) f_0^i(Q). \end{aligned} \quad (5.22)$$

The upper bound is  $X_k$  where

$$X_k = \sum_{j=0}^3 \sum_{i=0}^3 f_j(X_{k-1}^i) q_{ji} \pi_{k-1}^i \quad (5.23)$$

and  $X_{k-1}^i = E[X_{k-1} | \text{state is } i \text{ at time } (k-2)]$ .

A sufficient condition for divergence of expected error covariance is

$$q_{00} \cdot a > 1. \quad (5.24)$$

The equations for lower and upper bounds are pretty messy but they can be calculated iteratively. Also, these bounds are dependent on the value of  $q_{11}$  and initial distribution of packet-dropping  $\pi_0^i$ .

## 5.4 Simulation Results

In this section, simulation results are provided to verify the advantages of MD codes. We choose the discrete-time LTI system with  $A = -1.25$  and  $C = 1$ . Gaussian noises  $w_k$  and  $v_k$  have zero means and covariances  $G = 2.5$  and  $Q = 1$ , respectively. A balanced two-description MD code is designed such that the central distortion  $D_0 \approx 8.33 \times 10^{-6}$  and the side distortion  $D_1 \approx 1.56$ . The bpss of the MD code is 12 bits. At the same time, we choose a single description code as the counterpart with the same average distortion and the bpss is 10 bits.

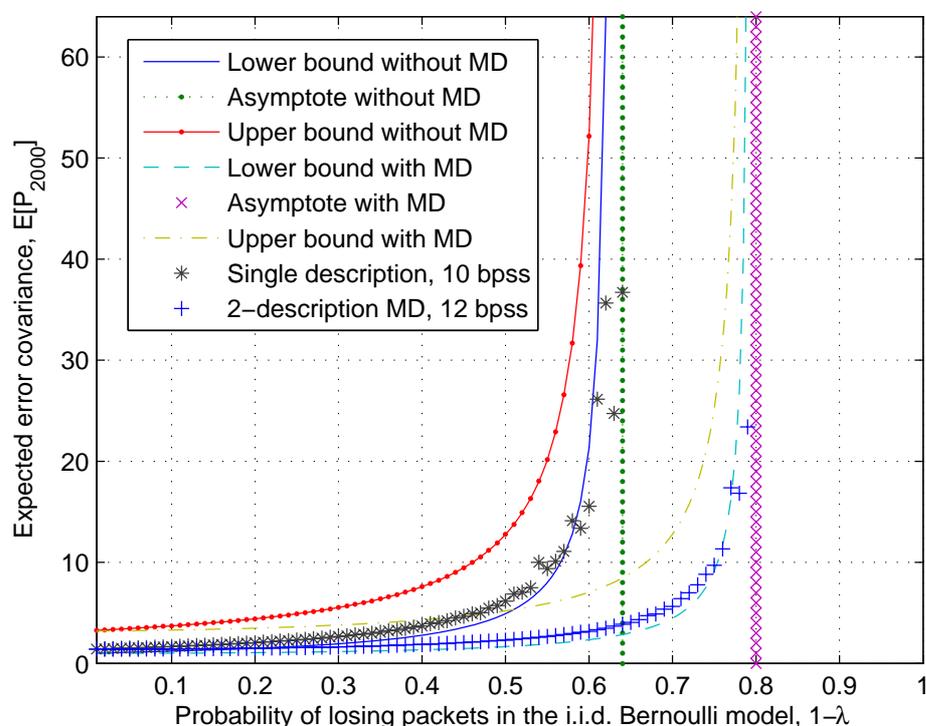


Figure 5.4: Simulation results of expected error covariances with theoretical upper and lower bounds

We start with the Bernoulli packet-dropping model. Figure 5.4 shows the simulation results of the expected error covariance. The theoretical upper and lower bounds with or without MD codes are calculated according to Theorem 5.3.5 and reference [94], respectively. The simulations are run 1000 times and each simulation is run 2000 time steps. We use the average value  $E[P_{2000}]$  as the expected error covariance. The asymptote  $\lambda_c$  is 0.36

for the single-description code and is 0.2 for the balanced two-description MD code. It is clear that convergence properties of error covariance at a high packet loss rate region are improved dramatically. Note that when  $\lambda$  is close to the asymptote, some of the simulated error covariances values are below the lower bound. The reason is that we only take limited time steps for the simulation, and a residual effect of the initial conditions remains.

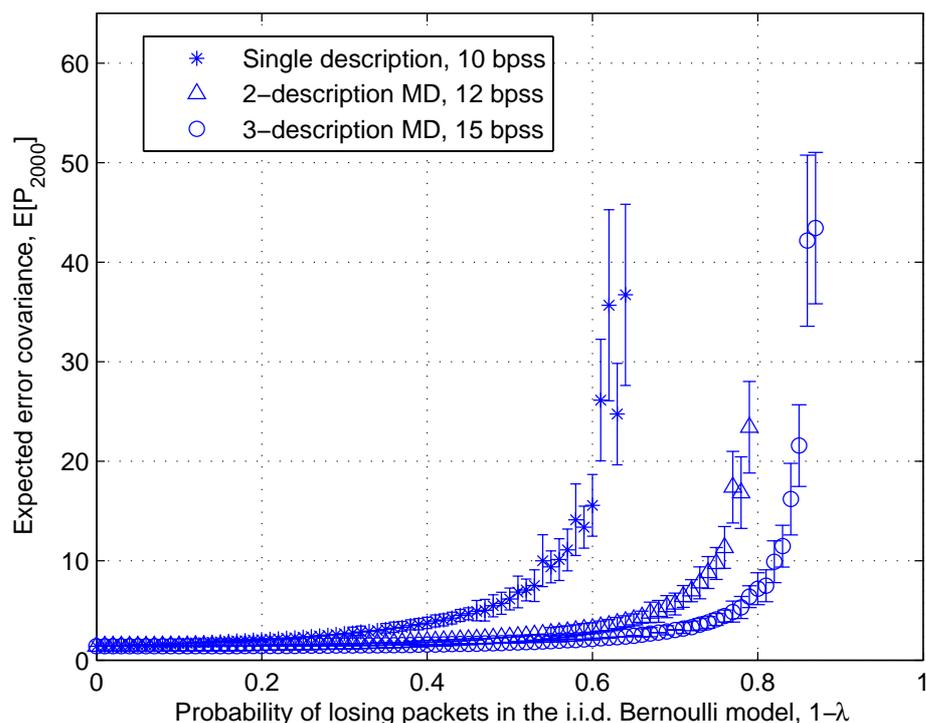


Figure 5.5: Mean values of error covariance with same central distortions

Figure 5.5 shows the different simulation results with different source codes. The center of the error bar is the mean value and 95% of the simulation results are located inside the error bars. It shows that, if we use a balanced three-description MD code, the critical value  $\lambda_c$  is even smaller since the choice of losing all descriptions of one observation is smaller, which is consistent with the results on the generalized MARE. So the benefits of using MD codes are clear and the cost we need to pay is more bits for each source sample. When we keep the bpss constant, as shown in Figure 5.6, we get larger quantization noise as the number of descriptions increases, but there are no obvious differences due to the relative high bpss in this example.

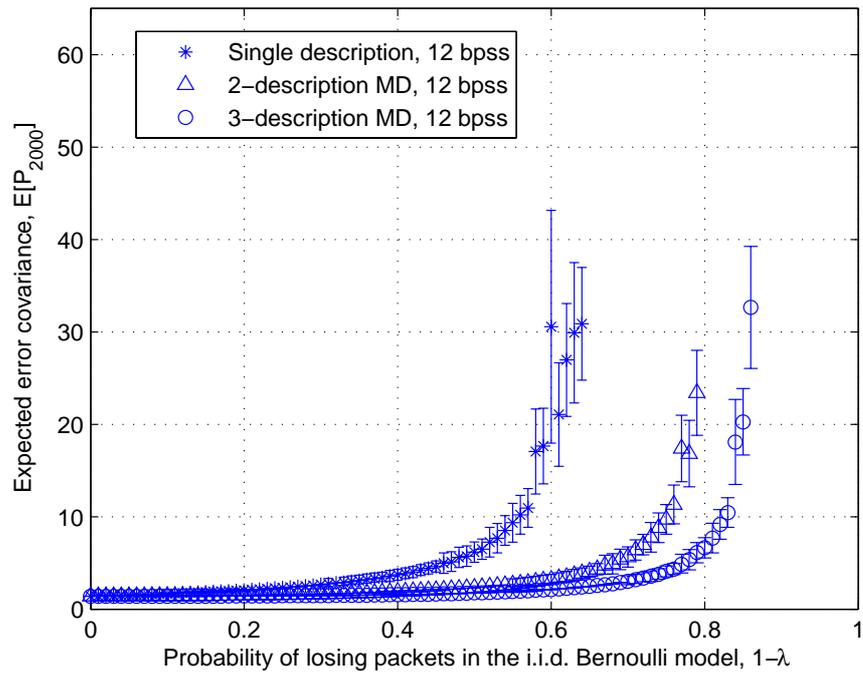


Figure 5.6: Mean values of error covariance with same bpss

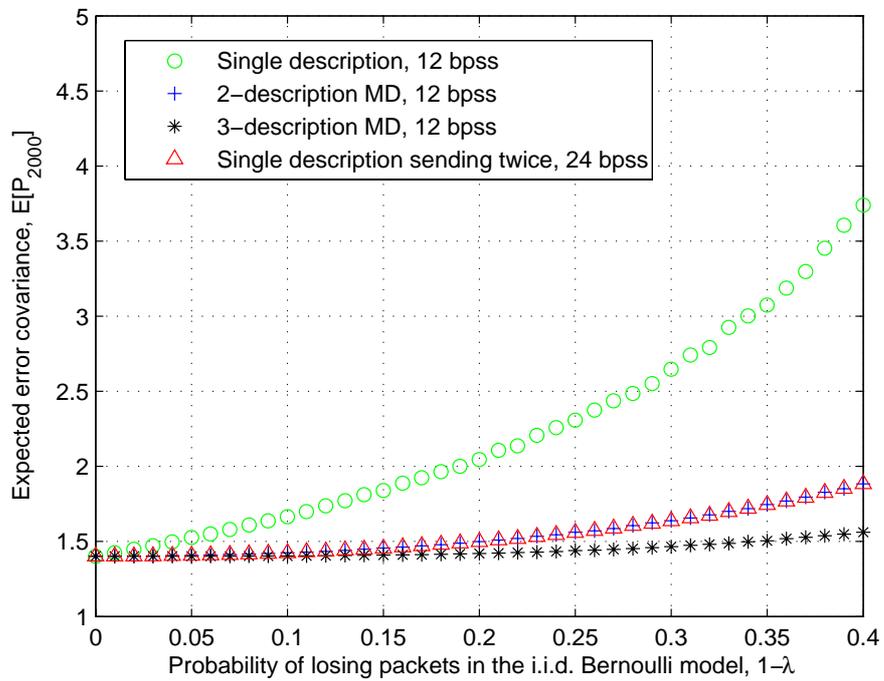


Figure 5.7: Mean values of error covariance with low dropping rate

Actually, the packet-dropping rate of a practical communication network is fairly small. Figure 5.7 shows the expected error covariance when the packet-dropping rate is low and MD codes give much better performance than the single-description code. Note that the two-description MD code achieves as good performance as sending the single-description code twice but saves up to 40% of the bandwidth.

Next, we use the two-state Markov chain model for packet drops. In Figure 5.8 we plot the theoretical bounds and simulation data for the error variance as a function of  $q_{10}$  under the conditions as  $q_{11} = 0.95$  and uniform distribution of  $\pi_0^i$ . The lowering of the bounds is indicative of the performance getting better with MD codes. The simulation results with parameters  $q_{11} = 0.05$  and  $q_{11} = 0.95$  with different coding schemes are shown in Figure 5.9.

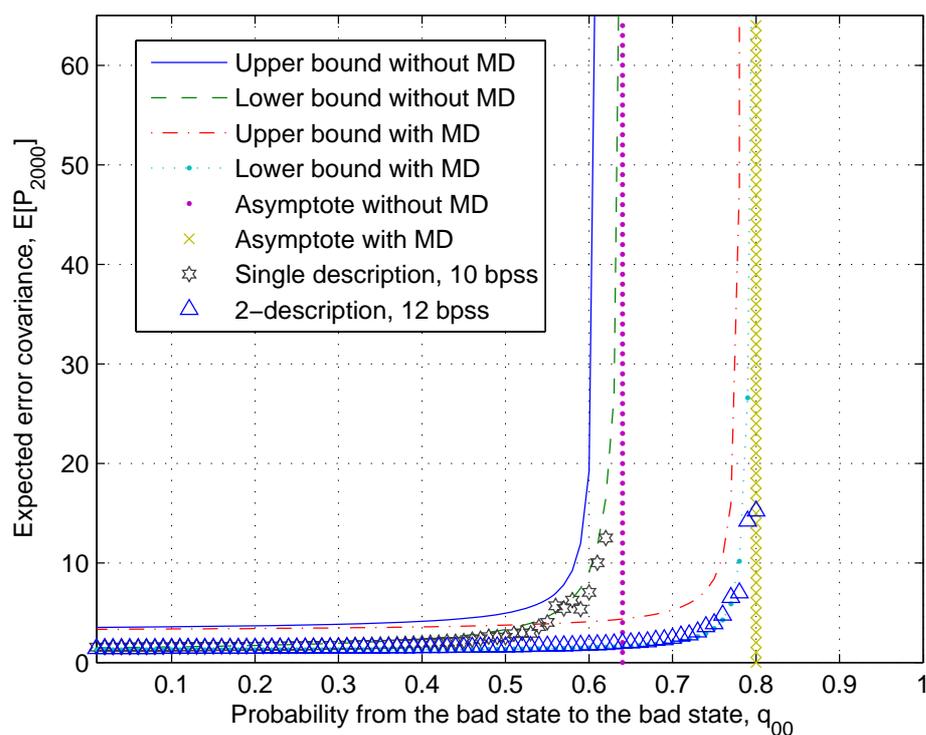


Figure 5.8: Theoretical upper and lower bounds for burst packet-dropping case with  $q_{11} = 95\%$

In the Markov chain model,  $q_{01} = 1 - q_{11}$  is a measure of how often the bursty dropping happens while  $q_{00} = 1 - q_{10}$  indicates how large the burst dropping is. According to

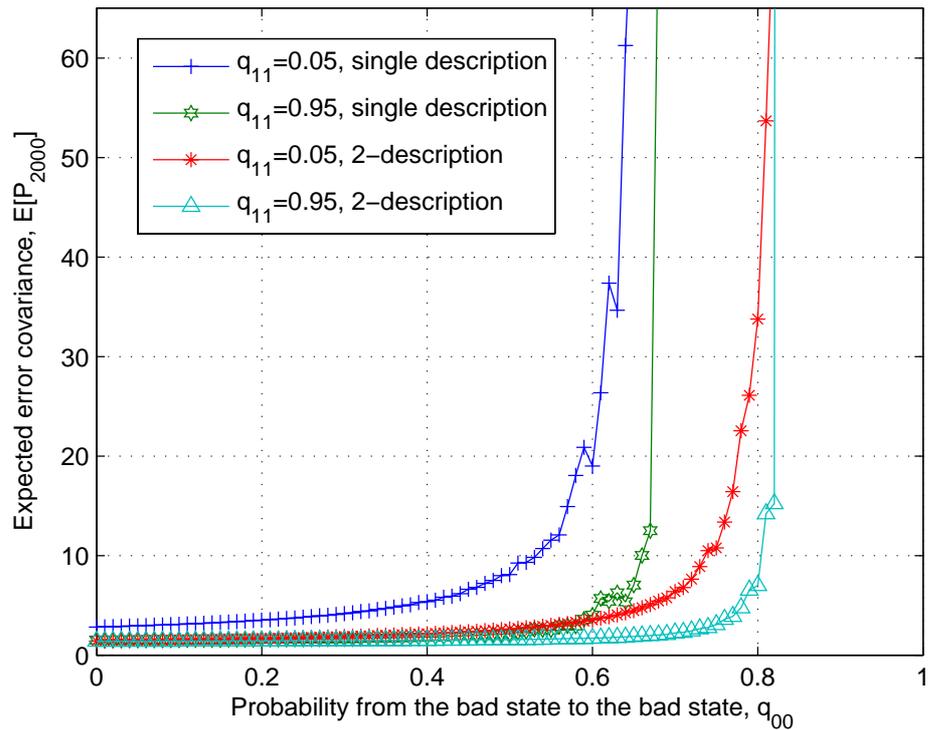


Figure 5.9: Simulation results for burst packet-dropping case

simulation results, expected error covariance diverges more quickly with higher  $q_{01}$ . This makes sense since if the error bursts happen more often, the estimated error covariance is expected to diverge more easily. It is seen from the figures that the system diverges around  $q_{10} = 0.36$  with the single-description code case and diverges around  $q_{10} = 0.2$  with the two-description MD code. Thus the stability margin is enlarged if we use MD codes. Also, for the same  $q_{10}$ , using MD codes greatly decreases the expected error covariance.

## Chapter 6

# Conclusions and Future Work

### 6.1 Summary of Main Contributions

In this work, we study coordinated control and state estimation of networked multi-agent systems. A novel distributed formation control strategy, the double-graph control strategy, is proposed where objective synchronization is processed in global coordination topology and directed interactions among agents are presented in local interaction topology. Using this strategy, each agent adjusts its behavior according to the global objective and the local environment (neighbors) simultaneously. A single parameter  $\alpha$  is used to balance these two incentives.

Suppose that all agents are synchronized on the coordination objective, the advantages of the double-graph control strategy are addressed from two aspects:

- For the stability issue, with identical local controller structure, Nyquist critical points are moved into an ellipse and the stable margin is enlarged. When the interaction topology is acyclic, the stability of a single agent implies the stability of the whole system. When it is not, the system is guaranteed to be stable as long as the infinity norm of the local transfer function is bounded where the upper bound relies on the size of the ellipse. Moreover, the stable condition is robust against the interaction topology.
- For the performance issue, we focus on disturbance resistance. It has been shown that the disturbance introduced at any agent will not be amplified when it propagates

through the formation. An upper bound for the amount of disturbance resistance, which is independent of the number of the agents, is given. When the interaction topology has one vertex as the leader, a more precise upper bound is given. Collision avoidance and time delays in interaction topology are also briefly discussed.

In the global coordination topology, we focus on the consensus protocol for objective synchronization. Based on the graph reduction on strong components, necessary and sufficient conditions for consensus seeking are given. This approach gives us more insights for this collective dynamics and the value of final consensus state is also presented explicitly. Also, multi-hop relay protocols are proposed for fast consensus seeking. We emphasize the two-hop relay protocol over connected and symmetric topologies for average consensus seeking. It has been shown that using relay protocols can efficiently improve the convergence speed without physically changing the topology. But the cost we need to pay is the extra communication bandwidth. Furthermore, a trade off between the convergence performance and robustness against communication time delays is identified. The more edges the graph includes, the faster the convergence speed is, while the more sensitive the protocol is to the time delay. We use the frequency sweep method to efficiently find the delay margin for two-hop relay protocols and verify this tradeoff on three typical communication topologies.

For packet-based state estimation in networked multi-agent systems, we use multiple description coding scheme to counteract the effect of packet drops. The accuracy of observation data received at the estimator only depends on how many descriptions are successfully transmitted. We considered two typical network packet drops models: the i.i.d. Bernoulli model and the Markov chain model. For the i.i.d. Bernoulli model, the expected value of error covariance is formulated as the solution of the MARE. Using MD codes, the convergence region of the estimation error covariance is much larger than using a traditional single-description code because the chance of losing all descriptions is much smaller. Boundaries for expected error covariance are also given. For the two-state Markov chain model, similar benefits are also shown in analytic results. Besides, MD codes are optimal codes that save considerably more bandwidth than sending duplicated packets. Thus, the impact of communication constraints can be understood from another angle: In high bit-

rate scenario and with large, complex communication networks, we compensate the packet loss and satisfy the real-time demands by using elegant coding schemes.

## 6.2 Future Directions and Possible Extensions

There are several directions and possible related research areas in which we can carry out future work.

**Optimal configuration for double-graph control strategy.** Design choices of double-graph control strategy include adjusting the value of  $\alpha$  and choosing appropriate transfer functions for a given interaction topology. For different applications, the double-graph strategy provides more flexibility and redundancy than previous approaches. Searching for the optimal configuration can be the next interesting research topic.

Another possible extension is the stability of interaction topology switching. This is useful for formation splitting/merging in order to overcome big obstacles or go through small open areas. Even though the formation is stable for each individual topology, inappropriate switching sequence may result instability. Necessary conditions for smooth and safe topology switching need to be specified.

**Robustness of Consensus Protocols.** Besides communication delays, there exist a couple of other issues related to the robustness of consensus protocols. For example, if there exist malicious agents among a networked multi-agent system who keeps sending out wrong messages, the consensus protocol may not be able to converge or converge to an unexpected value. It will be very useful if we can have an efficient mechanism to identify and isolate these “bad” agents so that we can synchronize other “healthy” agents.

Another interesting problem is studying consensus protocols with packet loss. It has been shown that those protocols still work when the network switches its topology as long as the union of these “infinity often” topologies contains a spanning tree strongly connected [62, 67]. Thus, we can expect that, with packet loss, consensus protocols still achieve the consensus, but need longer time to converge. The quantitative analysis needs to be investigated carefully.

**Collective protocols for networked multi-agent systems.** For coordination and coop-

eration in networked multi-agent systems, generalized distributed decision-making methods still remain unknown. The decision-making mechanism depends on the substance of the global objective. One possible formulation for this process is the following generalized discrete-time protocol

$$x_i(k+1) = f(x_i(k), x_j(k))$$

where  $j \in \mathcal{N}(i)$  and the function  $f(\cdot)$  is a selected collective function. When the output of  $f(\cdot)$  is the average of the inputs, this protocol is the discrete-time consensus protocol. Another simple example for collective functions is  $\max(\cdot)$ .

The connection between simple collective functions and a final group decision can be studied with respect to the connectivity of the communication topology. New mathematical tools are needed to formulate this connection. Given a communication topology and global objective, designing a collective function should be an interesting project. Similar to the consensus protocol, common performance issues include the convergence speeds, robustness against the topology, and the sensitivity to communication delays.

Except for the objective synchronization in formation control, coordination/cooperation tasks that can be solved by this framework include role section among connected groups and complex task assignment for nonhomogeneous robots/UAVs/underwater vehicles. More generally, this formation can be useful for self-organizing or self-assembling systems in physics, biology, and sociology.

**Packet-based state estimation for nonlinear dynamical systems.** A natural extension for packet-based estimation is estimating nonlinear systems. There exist some popular strategies for on-line nonlinear estimation, such as the extended Kalman filter (EKF) and moving horizon estimation (MHE). In today's networked environment, it is important to understand the performance of nonlinear estimation schemes with intermittent observation.

For EKF, the implementation is easy by linearizing the system. It has been used widely in chemical engineering. Theoretical properties of EKF have been explored extensively. With packet drops, stochastic convergence behaviors for error covariance should depend on the packet drop rate as well as Lipschitz constants for the nonlinear dynamics.

On the other hand, MHE is an approach for online state estimation problem with non-

linear discrete-time dynamics, constrained variables, and nonquadratic cost. Thus, it has more applications than EKF. If the arrival cost is known exactly, then MHE provides the optimal Bayesian estimate. In practice, the arrival cost is difficult to compute and must be approximated. Thus, the optimality cannot be guaranteed so far. Packet drops will affect the formulation of the constraints in MHE. Thus, finding the best scheme for rearranging observation constraints within a given window size should help to improve MHE.

Overall, MD codes should be able to improve the estimation even with nonlinear dynamics as well. Quantitative analysis needs to be conducted carefully to verify this conjecture.

## Bibliography

- [1] N. R. Sandell, P. Varaiya, M. Athans, and M. G. Safonov, "Survey of decentralized control methods for large scale systems," *IEEE Transactions on Automatic Control*, vol. 23, no. 2, pp. 108–128, 1978.
- [2] E. J. Davison, "A method for simplifying linear dynamic systems," *IEEE Transactions on Automatic Control*, vol. 11, no. 1, pp. 93–101, 1966.
- [3] M. Aoki, "Control of large-scale dynamic systems by aggregation," *IEEE Transactions on Automatic Control*, vol. 13, no. 3, pp. 246–253, Jun. 1968.
- [4] H. K. Ramapriyan, "Study of coupling in interconnected systems," Ph.D. dissertation, University of Minnesota, Minneapolis, 1970.
- [5] S. H. Wang and E. J. Davison, "On the stabilization of decentralized control systems," *IEEE Transactions on Automation Control*, vol. 18, no. 5, pp. 473–478, Oct. 1973.
- [6] J. P. Corfmat and A. S. Morse, "Decentralized control of linear multivariable systems," *Automatica*, vol. 12, pp. 479–495, 1976.
- [7] A. N. Michel and R. K. Miller, *Qualitative Analysis of Large Scale Dynamical Systems*, ser. Mathematics in Science and Engineering. Academic Press, 1978, vol. 134.
- [8] D. D. Šiliak, *Decentralized Control of Complex Systems*, ser. Mathematics in Science and Engineering. Academic Press, 1990, vol. 184.

- [9] E. J. Lasley and A. N. Michel, "Input-output stability of interconnected systems," *IEEE Transactions on Automatic Control*, vol. 21, no. 1, pp. 84–89, Feb. 1976.
- [10] P. A. Cook, "On the stability of interconnected systems," *International Journal of Control*, vol. 20, no. 3, pp. 407–415, 1974.
- [11] D. W. Porter and A. N. Michel, "Input-output stability of time-varying nonlinear multiloop feedback systems," *IEEE Transactions on Automatic Control*, vol. 19, no. 4, pp. 422–427, Aug. 1974.
- [12] M. Araki, "Input-output stability of composite feedback systems," *IEEE Transactions on Automatic Control*, vol. 21, no. 2, pp. 254–259, Apr. 1976.
- [13] K. Lau, S. Lichten, L. Young, and B. Haines, "An innovative deep space application of GPS technology of formation flying spacecraft." San Diego, CA: AIAA Guidance, Navigation and Control Conference, Jul. 1996.
- [14] V. Kapilal, A. G. Sparks, J. M. Buffington, and Q. Yan, "Spacecraft formation flying: dynamics and control." American Control Conference, 1999.
- [15] D. F. Chichka, "Satellite clusters with constant apparent distribution," *AIAA Journal of Guidance, Control, and Dynamics*, vol. 24, no. 1, pp. 117–122, 2001.
- [16] C. Sabol, R. Burns, and C. A. McLaughlin, "Satellite formation flying design and evolution," *AIAA Journal of Spacecraft and Rockets*, vol. 38, no. 2, pp. 270–278, 2001.
- [17] H. Schaub, S. R. Vadali, J. L. Junkins, and K. T. Alfriend, "Spacecraft formation flying control using mean orbit elements," *Journal of the Astronautical Sciences*, vol. 48, no. 1, pp. 69–87, Mar. 2000.
- [18] R. W. Beard, J. Lawton, and F. Y. Hadaegh, "A coordination architecture for spacecraft formation control," *IEEE Transactions on Control Systems Technology*, vol. 9, no. 6, pp. 777–790, Nov. 2001.

- [19] H. Yeh, E. Nelson, and A. Sparks, “Nonlinear tracking control for satellite formations,” *AIAA Journal of Guidance, Control, and Dynamics*, vol. 25, no. 2, pp. 376–386, 2002.
- [20] J. G. Bender, “An overview of systems studies of automated highway systems,” *IEEE Transactions on Vehicular Technology*, vol. 40, no. 1, pp. 82–99, 1991.
- [21] S. E. Shladover, C. A. Desoer, J. K. Hedrick, M. Tomizuka, J. Walrand, W. B. Zhang, D. H. McMahon, H. Peng, S. Sheikholeslam, and N. McKeown, “Automated vehicle control developments in the path program,” *IEEE Transactions on Vehicular Technology*, vol. 40, no. 1, pp. 114–130, Feb. 1991.
- [22] D. Swaroop and J. K. Hedrick, “String stability of interconnected systems,” *IEEE Transactions on Automatic Control*, vol. 41, no. 3, pp. 349–357, 1996.
- [23] ———, “Constant spacing strategies for platooning in automated highway systems,” *Journal of Dynamic Systems Measurement and Control-Transactions*, vol. 121, no. 3, pp. 462–470, 1999.
- [24] J. Eyre, D. Yanakiev, and I. Kanellakopoulos, “A simplified framework for string stability analysis of automated vehicles,” *Vehicle System Dynamics*, vol. 30, no. 5, pp. 375–405, Nov. 1998.
- [25] A. Pant, P. Seiler, and J. K. Hedrick, “Mesh stability of look-ahead interconnected systems,” *IEEE Transactions on Automatic Control*, vol. 47, no. 2, pp. 403–407, 2002.
- [26] T. W. McLain, P. R. Chandler, S. Rasmussen, and M. Pachter, “Cooperative control of UAV rendezvous,” in *Proceedings of the 2001 American Control Conference*, vol. 3, Arlington, VA, 2001, pp. 2309–2314.
- [27] F. Giulletti, L. Pollini, and M. Innocenti, “Autonomous formation flight,” *IEEE Control Systems Magazine*, vol. 20, no. 6, pp. 34–44, Dec. 2000.

- [28] S.-M. Li, J. D. Boskovic, S. S. nad R. Amin, J. Mehra, R. K. Mehra, R. W. Beard, and T. W. McLain, "Autonomous hierarchical control of multiple unmanned combat air vehicles (UCAVs)," in *Proceedings of the 2002 American Control Conference*, vol. 1, 2002, pp. 274–279.
- [29] T. B. Curtin, J. G. Bellingham, J. Catipovic, and D. Webb, "Autonomous oceanographic sampling networks," *Oceanography*, vol. 6, no. 3, pp. 86–94, 1993.
- [30] P. Bhatta, E. Fiorelli, F. Lekien, N. E. Leonard, D. Paley, F. Zhang, R. Bachmayer, R. E. Davis, D. M. Fratantoni, and R. Sepulchre, "Coordination of an underwater glider fleet for adaptive ocean sampling," in *Proc. International Workshop on Underwater Robotics*, Genoa, Italy, 2005.
- [31] E. Fiorelli, N. E. Leonard, P. Bhatta, D. Paley, R. Bachmayer, and D. M. Fratantoni, "Multi-AUV control and adaptive sampling in Monterey Bay," *IEEE Journal of Oceanic Engineering*, 2006, to appear.
- [32] C. Tomlin, G. J. Pappas, and S. Sastry, "Conflict resolution for air traffic management: A study in multiagent hybrid systems," *IEEE Transactions on Automatic Control*, vol. 43, no. 4, pp. 509–521, Apr. 1998.
- [33] R. L. Raffard, S. L. Waslander, A. M. Bayen, and C. J. Tomlin, "Cooperative distributed network control: Application to air traffic management," in *Proceedings of AIAA Guidance Navigation and Control*, 2005.
- [34] F. Paganini, Z. Wang, J. C. Doyle, and S. H. Low, "Congestion control for high performance, stability and fairness in general networks," *IEEE/ACM Transactions on Networking*, vol. 13, no. 1, pp. 43–56, Feb. 2005.
- [35] D. Estrin, R. Govindan, J. Heidemann, and S. Kumar, "Next century challenges: scalable coordination in sensor networks," in *Proceedings of the 5th annual ACM/IEEE international conference on Mobile computing and networking*, Seattle, WA, 1999, pp. 263–270.

- [36] J. A. Stankovic, T. E. Abdelzaher, C. Lu, S. Lui, and J. C. Hou, "Real-time communication and coordination in embedded sensor networks," in *Proceedings of the IEEE*, vol. 91, no. 7, 2003, pp. 1002–1022.
- [37] L. Xiao, S. Boyd, and S. Lall, "A scheme for robust distributed sensor fusion based on average consensus." International Conference on Information Processing in Sensor Networks, Apr. 2005, pp. 63–70.
- [38] R. M. Murray, Ed., *Control in an information rich world: report of the Panel on Future Directions in Control, Dynamics, and Systems*. SIAM, 2003. [Online]. Available: <http://www.cds.caltech.edu/~murray/cdspanel/>
- [39] M. A. Lewis and K. Tan, "High precision formation control of mobile robots using virtual structures," *Autonomous Robots*, vol. 4, no. 4, pp. 387–403, Oct. 1997.
- [40] T. Balch and R. C. Arkin, "Behavior-based formation control for multirobot teams," *IEEE Transactions on Robotics and Automation*, vol. 14, no. 6, pp. 926–939, Dec. 1998.
- [41] J. P. Desai, J. P. Ostrowski, and V. Kumar, "Modeling and control of formations of nonholonomic mobile robots," *IEEE Transactions on Robotics and Automation*, vol. 17, no. 6, pp. 905–908, Dec. 2001.
- [42] J. T. Feddema, C. Lewis, and D. A. Schoenwald, "Decentralized control of cooperative robotic vehicles: Theory and application," *IEEE Transactions on Robotics and Automation*, vol. 18, no. 5, pp. 852–864, Oct. 2002.
- [43] H. G. Tanner, G. J. Pappas, and V. Kumar, "Leader-to-formation stability," *IEEE Transactions on Robotics and Automation*, vol. 20, no. 3, pp. 443–455, Jun. 2004.
- [44] C. J. Schumacher and R. Kumar, "Adaptive control of UAVs in close-coupled formation flight," in *Proceedings of the 2000 American Control Conference*, vol. 2, 2000, pp. 849–853.

- [45] T. R. Smith, H. Hansmann, and N. E. Leonard, "Orientation control of multiple underwater vehicles with symmetry-breaking potentials," in *Proceedings of the 40th IEEE Conference on Decision and Control*, vol. 5, Orlando, FL, 2001, pp. 4598–4603.
- [46] J. A. Fax and R. M. Murray, "Information flow and cooperative control of vehicle formations," *IEEE Transactions on Automatic Control*, vol. 49, no. 9, pp. 1465–1476, Sep. 2004.
- [47] P. K. C. Wang, "Navigation strategies for multiple autonomous mobile robots moving in formation," *Journal of Robotics Systems*, vol. 8, no. 2, pp. 177–195, 1991.
- [48] M. Egerstedt, X. Hu, and A. Stotsky, "Control of mobile platforms using a virtual vehicle approach," *IEEE Transactions on Automatic Control*, vol. 46, no. 11, pp. 1777–1782, 2001.
- [49] N. E. Leonard and E. Fiorelli, "Virtual leaders, artificial potentials and coordinated control of groups," in *Proceedings of the 40th IEEE Conference on Decision and Control*, vol. 3, Orlando, FL, 2001, pp. 2968–2973.
- [50] W. Ren, "Consensus seeking, formation keeping, and trajectory tracking in multiple vehicle cooperative control," Ph.D. dissertation, Brigham Young University, 2004.
- [51] J. R. Lawton and R. W. Beard, "Synchronized multiple spacecraft rotations," *Automatica*, vol. 38, no. 8, pp. 1359–1364, 2002.
- [52] R. Diestel, *Graph Theory*, 3rd ed., ser. Graduate Texts in Mathematics. Springer-Verlag, 2005, vol. 173.
- [53] F. Callier, C. Wan, and C. Desoer, "Input-output stability theory of interconnected systems using decomposition techniques," *IEEE Transactions on Circuits and Systems*, vol. 23, no. 12, pp. 714–729, Dec. 1976.
- [54] J. F. Heagy, T. L. Carroll, and L. M. Pecora, "Synchronous chaos in coupled oscillator systems," *Physical Review E*, vol. 50, no. 3, pp. 1874–1885, 1994.

- [55] L. M. Pecora and T. L. Carroll, “Master stability functions for synchronized coupled systems,” *Phys. Rev. Lett.*, vol. 80, no. 10, pp. 2109–2112, 1998.
- [56] C. W. Wu and L. O. Chua, “Synchronization in an array of linearly coupled dynamical systems,” *IEEE Transactions on Circuits and Systems I: Fundamental Theory and Applications*, vol. 42, no. 8, pp. 430–447, Aug. 1995.
- [57] E. Shaw, “Fish in schools,” *Natural History*, vol. 84, no. 8, pp. 40–45, 1975.
- [58] A. Okubo, “Dynamical aspects of animal grouping: swarms, schools, flocks and herds,” *Advances in Biophysics*, vol. 22, pp. 1–94, 1986.
- [59] T. Vicsek, A. Czirok, E. Ben-Jacob, I. Cohen, and O. Schochet, “Novel type of phase transitions in a system of self-driven particles,” *Physical Review Letters*, vol. 75, pp. 1226–1229, 1995.
- [60] J. Toner and Y. Tu, “Flocks, herds, and schools: A quantitative theory of flocking,” *Physical Review E*, vol. 58, no. 4, pp. 4828–4858, 1998.
- [61] H. Levine, W. J. Rappel, and I. Cohen, “Self-organization in systems of self-propelled particles,” *Physical Review E*, vol. 63, p. 017101, 2001.
- [62] A. Jadbabaie, J. Lin, and A. S. Morse, “Coordination of groups of mobile autonomous agents using nearest neighbor rules,” *IEEE Transactions on Automatic Control*, vol. 48, no. 6, pp. 988–1001, Jun. 2003.
- [63] R. A. Horn and C. R. Johnson, *Matrix Analysis*. Cambridge University Press, 1999.
- [64] R. Rlfati-Saber and R. M. Murray, “Consensus problems in networks of agents with switching topology and time-delays,” *IEEE Transactions on Automatic Control*, vol. 49, no. 9, pp. 1520–1533, Sep. 2004.
- [65] Z. Lin, M. Broucke, and B. Francis, “Local control strategies for groups of mobile autonomous agents,” *IEEE Transactions on Automatic Control*, vol. 49, no. 4, pp. 622–629, 2004.

- [66] Z. Lin, B. Francis, and M. Maggiore, “Necessary and sufficient graphical conditions for formation control of unicycles,” *IEEE Transactions on Automatic Control*, vol. 50, no. 1, pp. 121–127, 2005.
- [67] W. Ren and R. W. Beard, “Consensus seeking in multi-agent systems under dynamically changing interaction topologies,” *IEEE Transactions on Automatic Control*, vol. 50, no. 5, pp. 655–661, May 2005.
- [68] L. Moreau, “Stability of multiagent systems with time-dependent communication links,” *IEEE Transactions on Automatic Control*, vol. 50, no. 2, pp. 169–182, Feb. 2005.
- [69] J. Lin, A. S. Morse, and B. D. O. Anderson, “The multi-agent rendezvous problem,” in *Proceedings of the 42nd IEEE Conference on Decision and Control*, vol. 2, 2003, pp. 1508–1513.
- [70] D. Bauso, L. Giarre, and R. Pesenti, “Distributed consensus protocols for coordinating buyers,” in *Proceedings of the 42nd IEEE Conference on Decision and Control*, vol. 1, 2003, pp. 588–592.
- [71] M. Mehyar, D. Spanos, J. Pongsajapan, S. Low, and R. Murray, “Distributed averaging on asynchronous communication networks,” in *Proceedings of the 44th IEEE Conference on Decision and Control, and European Control Conference*, Seville, Spain, 2005, pp. 7446–7451.
- [72] A. Rodriguez-Angeles and H. Nijmeijer, “Cooperative synchronization of robots via estimated state feedback,” in *Proceedings of the 42nd IEEE Conference on Decision and Control*, vol. 2, 2003, pp. 1514–1519.
- [73] R. Olfati-Saber, “Ultrafast consensus in small-world networks,” in *Proceeding of 2005 American Control Conference*, Jun. 2005, pp. 2371–2378.
- [74] E. Altman, T. Basar, and R. Srikant, “Congestion control as a stochastic control problem with action delays,” *Automatica*, vol. 35, pp. 1937–1950, Dec. 1999.

- [75] G. C. Walsh, H. Ye, and L. G. Bushnell, “Stability analysis of networked control systems,” *IEEE Transactions on Control Systems Technology*, vol. 10, no. 3, pp. 438–446, May 2002.
- [76] W. Zhang, M. S. Branicky, and S. M. Phillips, “Stability of networked control systems,” *IEEE Control Systems Magazine*, vol. 21, no. 1, pp. 84–99, Feb. 2001.
- [77] A. S. Matveev and A. V. Savkin, “The problem of state estimation via asynchronous communication channels with irregular transmission times,” *IEEE Trans. Automat. Contr.*, vol. 48, no. 4, pp. 670–676, Apr. 2003.
- [78] W. S. Wong and R. W. Brockett, “Systems with finite communication bandwidth constraints – part I: State estimation problems,” *IEEE Transaction on Automatic Control*, vol. 42, no. 9, pp. 1294–1299, Sep. 1997.
- [79] —, “Systems with finite communication bandwidth constraints – part II: Stabilization with limited information feedback,” *IEEE Transactions on Automatic Control*, vol. 44, no. 5, pp. 1049–1053, May 1999.
- [80] S. K. Mitter, “Control with limited information: the role of systems theory and information theory,” *IEEE Information Theory Society Newsletter*, vol. 4, no. 50, pp. 122–131, 2000.
- [81] R. W. Brockett and D. Liberzon, “Quantized feedback stabilization of linear systems,” *IEEE Transactions on Automatic Control*, vol. 45, no. 7, pp. 1279–1289, Jul. 2000.
- [82] N. Elia and S. K. Mitter, “Stabilization of linear systems with limited information,” *IEEE Transaction on Automatic Control*, vol. 46, no. 9, pp. 1384–1400, Sep. 2001.
- [83] S. Tatikonda and S. K. Mitter, “Control under communication constraints,” *IEEE Transaction on Automatic Control*, vol. 49, no. 7, pp. 1056–1068, Jul. 2004.
- [84] —, “Control over noisy channels,” *IEEE Transaction on Automatic Control*, vol. 49, no. 7, pp. 1196–1201, Jul. 2004.

- [85] A. Sahai, “Any-time information theory,” Ph.D. dissertation, Massachusetts Institute of Technology, 2001.
- [86] L. Shi and R. M. Murray, “Towards a packet-based control theory - part 1: stabilization over a packet-based network,” in *Proceedings of the 2005 American Control Conference*, vol. 2, 2005, pp. 1251–1256.
- [87] R. F. Stengel, *Optimal Control and Estimation*. Dover Publications, Inc., 1994.
- [88] M. S. Grewal and A. P. Andrews, *Kalman Filtering: Theory and Practice Using MATLAB*, 2nd ed. Wiley-Interscience, 2001.
- [89] C. Rao, J. Rawlings, and D. Mayne, “Constrained state estimation for nonlinear discrete-time systems: Stability and moving horizon approximations,” *IEEE Transactions on Automatic Control*, vol. 48, no. 2, pp. 246–258, Feb. 2003.
- [90] N. Nahi, “Optimal recursive estimation with uncertain observation,” *IEEE Transactions on Information Theory*, vol. 15, no. 4, pp. 457–462, Apr. 1969.
- [91] M. Hadidi and S. Schwartz, “Linear recursive state estimators under uncertain observations,” *IEEE Transactions on Information Theory*, vol. 24, no. 6, pp. 944–948, Dec. 1979.
- [92] J. Nilsson, B. Bernhardsson, and B. Wittenmark, “Stochastic analysis and control of real-time systems with random time delays,” *Automatica*, vol. 34, no. 1, pp. 57–64, Jan. 1998.
- [93] O. Costa, “Stationary filter for linear minimum mean square error estimator of discrete-time Markovian jump systems,” *IEEE Transactions on Automatic Control*, vol. 47, no. 8, pp. 1351–1356, Aug. 2002.
- [94] B. Sinopoli, L. Schenato, M. Franceschetti, K. Poolla, M. I. Jordan, and S. S. Sastry, “Kalman filtering with intermittent observations,” *IEEE Transactions on Automatic Control*, vol. 49, no. 9, pp. 1453–1464, Sep. 2004.

- [95] X. Liu and A. J. Goldsmith, “Kalman filtering with partial observation losses.” Paradise Island, Bahamas: IEEE Conference on Decision and Control, Dec. 2004.
- [96] E. N. Gilbert, “Capacity of a burst-noise channel,” *Bell System Technical Journal*, vol. 39, pp. 1253–1265, Sep. 1960.
- [97] E. O. Elliott, “Estimates of error rates for codes on burst-noise channels,” *Bell System Technical Journal*, vol. 42, pp. 1977–1997, Sep. 1963.
- [98] M. Yajnik, S. Moon, J. Kurose, and D. Towsley, “Measurement and modelling of the temporal dependence in packet loss,” in *INFOCOM '99. Eighteenth Annual Joint Conference of the IEEE Computer and Communications Societies. Proceedings. IEEE*, vol. 1, New York, NY, Mar. 1999, pp. 345–352.
- [99] A. J. Fax, “Optimal and cooperative control of vehicle formations,” Ph.D. dissertation, California Institute of Technology, 2002.
- [100] N. Biggs, *Algebraic Graph Theory*, 2nd ed. Cambridge University Press, 1993.
- [101] C. Godsil and G. Royle, *Algebraic Graph Theory*, ser. Graduate Texts in Mathematics. Springer, 2001.
- [102] T. M. Cover and J. A. Thomas, *Elements of Information Theory*, ser. Wiley Series in Telecommunications. Wiley-Interscience, 1991.
- [103] S. P. Lloyd, “Least squares quantization in pcm,” *IEEE Transactions on Information Theory*, vol. 28, no. 2, pp. 129–137, Mar. 1982.
- [104] S. J. Mason, “Feedback theory – further properties of signal flow graphs,” *Proceedings of the IRE*, vol. 44, pp. 920–926, 1956.
- [105] P. J. Seiler, “Coordinated control of unmanned aerial vehicles,” Ph.D. dissertation, University of California, Berkeley, 2001.
- [106] J. Doyle, B. Francis, and A. Tannenbaum, *Feedback Control Theory*. Macmillan Publishing Co., 1990.

- [107] W. S. Levine, *The control handbook*. CRC press, 1996.
- [108] T. Chung, L. Cremean, W. Dunbar, Z. Jin, E. Klavins, D. M. and A. Tiwari, D. van Gogh, and S. Waydo, “A platform for cooperative and coordinated control of multiple vehicles: the Caltech multi-vehicle wireless testbed.” Conference on Cooperative Control and Optimization, Dec. 2003.
- [109] L. Xiao and S. Boyd, “Fast linear iterations for distributed averaging,” *Systems and Control Letters*, vol. 53, pp. 65–78, 2004.
- [110] K. Gu, V. L. Kharitonov, and J. Chen, *Stability of Time-Delay Systems*, ser. Control Engineering. Birkhauser, 2003.
- [111] W. R. Stevens, *TCP/IP Illustrated, Volume 1: The Protocols*, ser. Addison-Wesley Professional. Addison-Wesley Professional, 1993.
- [112] L. Ozarow, “On a source-coding problem with two channels and three receivers,” *Bell System Technical Journal*, vol. 59, no. 10, pp. 1909–1921, 1980.
- [113] V. A. Vaishampayan, “Design of multiple description scalar quantizers,” *IEEE Transactions on Information Theory*, vol. 39, no. 3, pp. 821–834, 1993.
- [114] D. Marco and D. L. Neuhoff, “The validity of the additive noise model for uniform scalar quantizers,” *IEEE Transactions on Information Theory*, vol. 51, no. 5, pp. 1739–1755, May 2005.
- [115] V. Gupta, T. H. Chung, B. Hassibi, and R. M. Murray, “On a stochastic sensor selection algorithm with applications in sensor scheduling and dynamic sensor coverage,” *Automatica*, vol. 42, no. 2, pp. 251–260, Feb. 2006.
- [116] G. E. Dullerud and F. Paganini, *A Course in Robust Control Theory: A Convex Approach*, ser. Texts in Applied Mathematics. Springer, 1999.
- [117] B. Friedland, *Control System Design: An Introduction to State-Space Methods*. McGraw-Hill, 1986.

- [118] H. K. Khalil, *Nonlinear Systems*, 3rd ed. Prentice Hall, 2002.
- [119] R. J. McEliece, *The Theory of Information and Coding*, 2nd ed., ser. Encyclopedia of Mathematics and Its Applications. Cambridge University Press, 2002.
- [120] S. Haykin, *Communication Systems*, 4th ed. John Wiley & Sons, Inc., 2001.
- [121] V. A. Vaishampayan and J. C. Batllo, “Asymptotic analysis of multiple description quantizers,” *IEEE Transactions on Information Theory*, vol. 44, no. 1, pp. 278–284, 1998.
- [122] M. Fleming, Q. Zhao, and M. Effros, “Network vector quantization,” *IEEE Transactions on Information Theory*, vol. 50, no. 8, pp. 1584–1604, 2004.
- [123] V. K. Goyal, “Multiple description coding: Compression meets the network,” *IEEE Signal Processing Mag.*, vol. 18, pp. 74–93, 2001.
- [124] R. Venkataramani, G. Kramer, and V. K. Goyal, “Multiple description coding with many channels,” *IEEE Transactions on Information Theory*, vol. 49, no. 9, pp. 2106–2114, 2003.
- [125] A. A. E. Gamal and T. M. Cover, “Achievable rates for multiple descriptions,” *IEEE Transaction on Information Theory*, vol. 28, no. 6, pp. 851–857, 1982.
- [126] R. Zamir, “Gaussian codes and Shannon bounds for multiple descriptions,” *IEEE Transaction on Information Theory*, vol. 45, no. 7, pp. 2629–2635, 1999.
- [127] D. Liberzon, “On stabilization of linear systems with limited information,” *IEEE Transaction on Automatic Control*, vol. 48, no. 2, pp. 304–307, Feb. 2003.
- [128] R. W. Brockett and D. Liberzon, “Quantized feedback stabilization of linear systems,” *IEEE Transaction on Automatic Control*, vol. 45, no. 7, pp. 1279–1289, Jul. 2000.
- [129] B. Sinopoli, C. Sharp, L. Schenato, S. Schaffert, and S. S. Sastry, “Distributed control applications within sensor networks,” *Proceedings of the IEEE*, vol. 91, no. 8, pp. 1235–1246, Aug. 2003.

- [130] N. Gordon, D. Salmond, and A. Smith, “A novel approach to nonlinear nongaussian Bayesian state estimation,” in *Proc. IEEE Conf. Radar and Signal Processing*, vol. 140, 1993.
- [131] L. J. Schulman, “Communication on noisy channels: A coding theorem for computation,” in *Proceedings of the 33rd Annual Symposium on Foundations of Computer Science*, 1992, pp. 724–733.
- [132] Z. Jin and R. M. Murray, “Stability and performance analysis with double-graph model of vehicle formations,” in *Proceedings of the 2003 American Control Conference*, vol. 3, 2003, pp. 2223–2228.
- [133] P. Tabuada, G. J. Pappas, and P. Lima, “Feasible formations of multi-agent systems,” in *Proceedings of the 2001 American Control Conference*, vol. 1, Arlington, VA, 2001, pp. 56–61.
- [134] M. Fleming and M. Effros, “Generalized multiple description vector quantization,” in *Proceedings of the data compression conference*, Snowbird, UT, Mar. 1999, pp. 3–12.
- [135] V. K. Goyal, J. Kovaceric, R. Aarean, and M. Vetterli, “Multiple description transform coding of images,” in *Proceedings of IEEE International Conference on Image Processing*, vol. 1, Chicago, IL, Oct. 1998, pp. 674–678.
- [136] Z. Jin and R. M. Murray, “Consensus controllability for coordinated multiple vehicle control.” Gainesville, FL: the 6th International Conference on Cooperative Control and Optimization, Feb. 2006.

## Appendix A

### Index Assignment Method for MDSQ

The “modified nested” index assignment method in [113] is a tractable and systematic method for designing balanced MDSQ index mapping matrices. From 1 to  $N$ , the index  $n$  is placed on the main diagonal and the  $2k$  diagonals that lie closest to the main diagonal, from left-upper corner to right-lower corner. Note that  $0 < k < \sqrt{M}$ , let  $k_i = \min(k, \sqrt{M} - i)$ . Define the action  $\bar{E}_i$  as putting index  $n$  into the sequence of matrix cells  $(i, i)$ ,  $(i, i + 1)$ ,  $(i + 1, i)$ ,  $(i, i + 2)$ ,  $(i + 2, i)$ ,  $\dots$ ,  $(i, i + k_i)$ ,  $(i + k_i, i)$  and the action  $\bar{S}_i$  as putting index  $n$  into the sequence of matrix cells  $(i, i)$ ,  $(i + 1, i)$ ,  $(i, i + 1)$ ,  $(i + 2, i)$ ,  $(i, i + 2)$ ,  $\dots$ ,  $(i + k_i, i)$ ,  $(i, i + k_i)$ . Figure A.1 shows these two different actions where part (a) is the action  $\bar{E}_i$  and part (b) is the action  $\bar{S}_i$ . From  $n = 1$  and matrix cell  $(i, i) = (1, 1)$ , we repeat action  $\bar{E}_i$   $k$  times, then repeat action  $\bar{S}_i$   $k$  times, then switch to action  $\bar{E}_i$  for  $k$  times again and continue this process until we reach the cell  $(\sqrt{M}, \sqrt{M})$ . Then we have

$$\begin{cases} N &= (2k + 1)\sqrt{M} - k(k + 1) \\ R &= R_i = R_j = \log_2(\sqrt{M}) \\ R_c &= \log_2(N) \end{cases} \quad (\text{A.1})$$

where  $R_c$  is the bps of the uniform quantizer in step one. Figure A.2 shows two simple examples of index mapping. In part (a),  $M = 64$  and  $k = 1$ ; while in part (b),  $M = 64$  and  $k = 2$ .

For given  $k$  and  $R$ , we use  $MN(R, k)$  to denote the MD code. Let  $\sqrt{M} = k^{1/\alpha}$  where  $0 < \alpha < 1$ . When  $M$  and  $N$  is large, i.e., at high bit rate, we can get these relationships

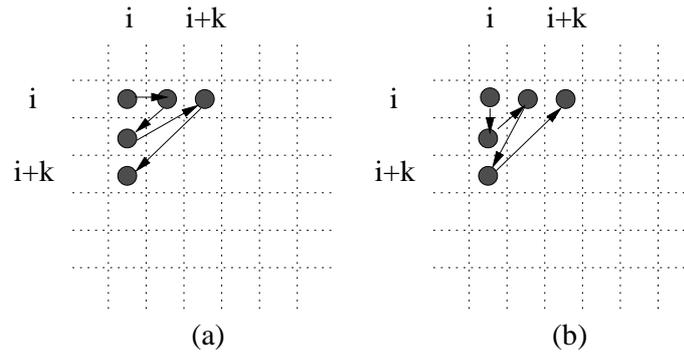


Figure A.1: Two actions to fill the index mapping matrix

about the central and side distortion:

$$\begin{cases} D_c \approx C_0 2^{-2R(1+\alpha)} \\ D_i \approx C_1 2^{-2R(1-\alpha)} \end{cases} \quad (\text{A.2})$$

where  $C_0$  and  $C_1$  are constants. There exist two extreme cases:

- When  $\alpha$  approaches 1,  $D_c$  decreases at the maximum rate of  $2^{-4R}$  while both  $D_i$  and  $D_j$  exhibit zero decay rates. In this case, the matrix is almost full and  $M \approx N$ ;
- When  $\alpha$  approaches 0,  $D_c$ ,  $D_i$  and  $D_j$  decay at the same rate of  $2^{-2R}$ . In this case, the matrix is only filled in the main diagonal.

1	2						
3	4	6					
	5	7	8				
		9	10	12			
			11	13	14		
				15	16	18	
					17	19	20
						21	22

1	2	4					
3	6	7	9				
5	8	11	13	15			
	10	12	16	18	20		
		14	17	21	22	24	
			19	23	26	27	29
				25	28	31	33
					30	32	34

Figure A.2: Examples of index mapping

Intermediate rates can be achieved by selecting other values of  $\alpha$ . When designing MD codes, we need to make sure that there are enough cells in these diagonals to hold the

outputs of the quantizer. We adjust the values of  $\sqrt{M}$  and  $k$  to control  $\alpha$ . For the simulation in Chapter 5, we choose  $MN(\log_2(40), 15)$  as the balanced two-description code.

## Appendix B

### Additional Proofs

#### Proof of Lemma 5.3.1

**Part (a)** According to the assumptions, we have

$$\begin{aligned}
 K_{X_0} &= -AXC^T(CXC^T + G_0)^{-1} \\
 \Rightarrow (A + KX_0C)XC^T + K_{X_0}G_0 &= 0 \\
 \Rightarrow F_{X_0}XC^T + K_{X_0}G_0 &= 0.
 \end{aligned}$$

Using the same reasoning, we get

$$\begin{aligned}
 K_{X_1} &= -AXC^T(CXC^T + G_1)^{-1} \\
 \Rightarrow F_{X_1}XC^T + K_{X_1}G_1 &= 0.
 \end{aligned}$$

Now the operator is

$$\begin{aligned}
 \phi(K_{X_0}, K_{X_1}, X) &= (1 - \lambda)^2(AXA^T + Q) + \lambda^2(F_0XF_0^T + V_0) + 2(1 - \lambda)\lambda(F_1XF_1^T + V_1) \\
 &= (1 - \lambda)^2(AXA^T + Q) + \lambda^2(F_0XF_0^T + K_{X_0}G_0K_{X_0}^T + Q) \\
 &\quad + 2(1 - \lambda)\lambda(F_1XF_1^T + K_{X_1}G_1K_{X_1}^T + Q) \\
 &= (1 - \lambda)^2(AXA^T + Q) + \lambda^2(F_0XA^T + F_0XC^TK_{X_0}^T + K_{X_0}G_0K_{X_0}^T + Q) \\
 &\quad + 2(1 - \lambda)\lambda(F_1XA^T + F_1XC^TK_{X_1}^T + K_{X_1}G_1K_{X_1}^T + Q)
 \end{aligned}$$

Recalling the equations of  $K_{X0}$  and  $K_{X1}$ , we have

$$\begin{aligned}\phi(K_{X0}, K_{X1}, X) &= (1 - \lambda)^2(AXA^T + Q) + \lambda^2(F_0XA^T + Q) + 2(1 - \lambda)\lambda(F_1XA^T + Q) \\ &= (1 - \lambda)^2(AXA^T + Q) + \lambda^2((A + K_{X0})XA^T + Q) \\ &\quad + 2(1 - \lambda)\lambda((A + K_{X1})XA^T + Q).\end{aligned}$$

Reorganizing the right side of the equation, we can simplify it as

$$\begin{aligned}\phi(K_{X0}, K_{X1}, X) &= (1 - \lambda)^2(AXA^T + Q) + \lambda^2(AXA^T + Q) - \lambda^2AXC^T(CXC^T + G_0)^{-1}CXA^T \\ &\quad + 2(1 - \lambda)\lambda(AXA^T + Q) - 2(1 - \lambda)\lambdaAXC^T(CXC^T + G_1)^{-1}CXA^T \\ &= AXA^T + Q - \lambda^2AXC^T(CXC^T + G_0)^{-1}CXA^T \\ &\quad - 2(1 - \lambda)\lambdaAXC^T(CXC^T + G_1)^{-1}CXA^T \\ &= g_\lambda(X).\end{aligned}$$

**Part (b)** The operator has three positive terms. The second one is a function of  $K_0$  and the third one is a function of  $K_1$ . Thus, if  $K_0$  minimizes the second term and  $K_1$  minimizes the third term, then  $(K_0, K_1)$  minimizes the whole function. We differentiate the second term and let it equal to zero.

$$\begin{aligned}2(A + K_0C)XC^T + 2K_0G_0 &= 0 \\ \Rightarrow K_0 &= -AXC^T(CXC^T + G_0)^{-1} = K_{X0}.\end{aligned}$$

Using the same method, we have

$$K_1 = -AXC^T(CXC^T + G_1)^{-1} = K_{X1}.$$

According to part (a), it is true that

$$g_\lambda(X) = \phi(K_{X0}, K_{X1}, X) = \min_{(K_0, K_1)} \phi(K_0, K_1, X).$$

**Part (c)**

$$g_\lambda(X) = \phi(K_{X0}, K_{X1}, X) \leq \phi(K_{X0}, K_{X1}, Y) \leq \phi(K_{Y0}, K_{Y1}, Y) = g_\lambda(Y).$$

**Part (d)** Since  $AXC^T(CXC^T + G_0)^{-1}CXA^T \geq 0$  and  $AXC^T(CXC^T + G_1)^{-1}CXA^T \geq 0$ , so if  $\lambda_1 \leq \lambda_2$ ,  $g_{\lambda_1}(X) \geq g_{\lambda_2}(X)$ .

**Part (e)** Let  $Z = \alpha X + (1 - \alpha)Y$ , we have

$$\begin{aligned} g_\lambda(Z) &= \phi(K_{Z_0}, K_{Z_1}, Z) \\ &= \alpha \cdot \phi(K_{Z_0}, K_{Z_1}, X) + (1 - \alpha)\phi(K_{Z_0}, K_{Z_1}, Y) \\ &\geq \alpha \cdot \phi(K_{X_0}, K_{X_1}, X) + (1 - \alpha)\phi(K_{Y_0}, K_{Y_1}, Y) \\ &= \alpha \cdot g_\lambda(X) + (1 - \alpha) \cdot g_\lambda(Y). \end{aligned}$$

This result shows that  $g_\lambda(X)$  is a concave function of  $X$  as long as  $X > 0$ .

**Part (f)**

$$\begin{aligned} g_\lambda(X) &= \phi(K_{X_0}, K_{X_1}, X) \\ &= (1 - \lambda)^2(AXA^T + Q) + \lambda^2(F_{X_0}XF_{X_0}^T + V_0) \\ &\quad + 2(1 - \lambda)\lambda(F_{X_1}XF_{X_1}^T + V_1) \\ &= (1 - \lambda)^2AXA^T + Q + \lambda^2(F_{X_0}XF_{X_0}^T + K_{X_0}G_0K_{X_0}^T) \\ &\quad + 2(1 - \lambda)\lambda(F_{X_1}XF_{X_1}^T + K_{X_1}G_1K_{X_1}^T) \\ &\geq (1 - \lambda)^2AXA^T + Q. \end{aligned}$$

**Part (g)** Let  $\hat{X}$  is the solution of the Lyapunov equation, i.e.,  $\hat{X} = (1 - \lambda)A\hat{X}(1 - \lambda)A^T + Q$ , then  $\hat{X} > 0$  and

$$\bar{X} - \hat{X} \geq (1 - \lambda)^2A(\bar{X} - \hat{X})A^T \geq 0.$$

**Part (h)** Since  $X$  is a random variable, then  $(1 - \lambda)^2AE[X]A^T + Q \leq E[g_\lambda(X)] \leq g_\lambda(E[X])$  where the first inequality comes from part (f) and the second one comes from part(e).

### Proof of Lemma 5.3.2

Please refer to [94] for details.

### Proof of Lemma 5.3.3

**Part (a)** First of all, we show that for all  $W \geq 0$ ,  $L(W) \geq 0$ . Also, if  $W_1 \leq W_2$ , then

$L(W_1) \leq L(W_2)$ . Choose  $0 \leq r < 1$  so that  $L(\bar{Y}) < r\bar{Y}$  and  $0 \leq m$  so that  $W \leq m\bar{Y}$ . Thus,

$$0 \leq L^k(W) \leq mL^k(\bar{Y}) \leq mr^k\bar{Y}.$$

When we take the limit  $k \rightarrow \infty$ , the result follows.

**Part (b)** The linear iteration is

$$\begin{aligned} Y_k &= L^k(Y_0) + \sum_{i=0}^{k-1} L^i(V) \\ &\leq \left(m_{Y_0} \cdot r^k + \sum_{i=0}^{k-1} m_V r^i\right) \bar{Y} \\ &\leq \left(m_{Y_0} \cdot r^k + \frac{m_V}{1-r}\right) \bar{Y} \\ &= \left(m_{Y_0} + \frac{m_V}{1-r}\right) \bar{Y}. \end{aligned}$$

#### Proof of Lemma 5.3.4

First, we define the matrices  $F_0 = A + K_0C$  and  $F_1 = A + K_1C$ . Consider the linear operator

$$L(Y) = (1 - \lambda)^2 AYA^T + \lambda^2 F_0 Y F_0^T + 2(1 - \lambda)\lambda F_1 Y F_1^T,$$

and observe that

$$\bar{P} > \phi(\bar{K}_0, \bar{K}_1, \bar{P}) = L(\bar{P}) + Q + \lambda^2 K_0 G_0 K_0^T + 2\lambda(1 - \lambda)K_1 G_1 K_1^T \geq \mathfrak{L}(\bar{P}).$$

Also, we have

$$P_{k+1} = g_\lambda(P_k) \leq \phi(\bar{K}_0, \bar{K}_1, P_t) = L(P_t) + V$$

where  $V = Q + \lambda^2 K_0 G_0 K_0^T + 2\lambda(1 - \lambda)K_1 G_1 K_1^T$ . Based on Lemma 5.3.3, the sequence  $P_t$  is bounded.