NEW APPROACHES TO DISTRIBUTED STATE ESTIMATION, INFERENCE AND LEARNING WITH EXTENSIONS TO BYZANTINE-RESILIENCE

A Dissertation

Submitted to the Faculty

of

Purdue University

by

Aritra Mitra

In Partial Fulfillment of the

Requirements for the Degree

of

Doctor of Philosophy

August 2020

Purdue University

West Lafayette, Indiana

THE PURDUE UNIVERSITY GRADUATE SCHOOL
STATEMENT OF DISSERTATION APPROVAL

Dr. Shreyas Sundaram, Chair

School of Electrical and Computer Engineering, Purdue University

Dr. Jianghai Hu

School of Electrical and Computer Engineering, Purdue University

Dr. Saurabh Bagchi

School of Electrical and Computer Engineering, Purdue University

Dr. Shaoshuai Mou

School of Aeronautics and Astronautics, Purdue University

Dr. Stanislaw H. Zak

School of Electrical and Computer Engineering, Purdue University

Dr. Paulo Tabuada

Department of Electrical and Computer Engineering, University of California at Los Angeles

Approved by:

Dr. Dimitrios Peroulis

Head of the School Graduate Program

To Baba, Maa, Chini, Mini,
and Nandini.

ACKNOWLEDGMENTS

It is often hard to put into words how the presence of certain individuals shapes the course of one's life. Other than my inherent laziness, perhaps this explains why I have been delaying the write up of this portion of my thesis. But the time has now come to take up this daunting task, and so here goes.

First, and foremost, I am indebted to my advisor, Shreyas Sundaram, for helping me complete this journey, and making it so enjoyable. I consider myself fortunate to have witnessed in person Shreyas's enthusiasm for research, his strong work ethics and ideals, his keen insights, and most of all, his ability to dissect complex problems. I greatly appreciate the calmness and patience with which he has taught me various things during the course of our weekly interactions. Over the years, his timely feedback has prevented my research trajectory from going astray and has led me to gain confidence in my own abilities. I am also grateful to him for the countless hours he has devoted to the improvement of my writing and presentation skills. In short, I could not have asked for a better doctoral advisor.

I would like to thank all my committee members: Professors Jianghai Hu, Saurabh Bagchi, Shaoshuai Mou, Stanislaw Zak, and Paulo Tabuada, for their valuable inputs. I have also thoroughly enjoyed collaborating on some of my research projects with Dr. John Richards from Sandia National Labs, and Professor Waseem Abbas from Vanderbilt University. As my life as a graduate student comes to an end, I am saddened by the thought of not being able to take new courses and attend lectures anymore. Nonetheless, during my stay at Purdue, I have had the pleasure of attending several amazing classes, of which, Professor Mark Bell's fascinating course on Information Theory is one I will never forget; I thank him, and all my other teachers. A heartfelt thank you goes out to the excellent staff of the ECE department who have worked behind the scenes, and efficiently filtered out all the administrative ripples, allowing me to concentrate exclusively on my research. In this regard, I am particularly grateful to Mary Ann; your oreo cakes and cookies will be dearly missed!

I thank all my lab mates Ashish, Amritha, Hemant, Lintao, Mustafa, Kananart, Russel, Nathan, Yanzhe, Lei, Tong, Nadia, and Maria, for helping maintain a congenial research atmosphere; I have learned a lot from our discussions. To Ashish, thank you for being a great senior, for your advice on various matters, and for the very interesting (and amusing) parallels between academia and cricket.

Now that I look back, I cannot imagine how the last 5 years would have turned out had I not come across some wonderful people. My first friend in West Lafayette was Sreya, who also cooked the first proper meal I had in the US. I will miss all our impromptu evening coffee breaks that were so refreshing. To Sayantan da, Sanchari, Prabudhya, Arindam, Indrani, Somrita, and Srishti, thank you for all the musical sessions together, the amazing potlucks, and the (rather short-lived) outdoor recreation stints. To Satyabrata, our friendship got strengthened while solving homework problems on every course we took together (a total of three, maybe?); I thank you and Kavya for being so cool and warm at the same time, and for introducing me to the best ramen place in West Lafayette: Yatagarasu. To Debasmit, Soumendu, and Dhruv, I will never forget our trip to Vegas and the Grand Canyon. To Debkalpa, Aniket and Rohan, I will really miss our traditional Happy Hollow walks on Durga Puja. To Ankush da, thank you for your friendship, all our laughs together, your (sometimes unsolicited) words of wisdom, and your excellent movie and mystery novel recommendations. To Esha di and Sayan da, I cannot thank you enough for your warmth and companionship, for all the delicious meals, and for those Friday evenings drenched with music, laughter, and fun board games. Sayan and Suman, thank you for being amazing roommates. Without all of you, I would not have grown so attached to West Lafayette. Finally, to Ratul, Aritra, Subhra, and Debaditya, thank you for tolerating me all these years, and for being just a phone call away.

Without the unconditional love and support of my parents Abhijit Mitra and Shrabani Mitra, none of my endeavors would have seen the light of day. I am grateful to them for instilling in me the value of education early on, and for helping me become the person I am. Despite being thousands of miles away, their words have been a constant source of encouragement for me to dream bigger, and move forward. Baba and Maa, since I am bereft of words to express my gratitude, I can only offer you this thesis as a humble token of my love and appreciation. My life would be incomplete without my two little sisters Ahona and Aheli. I love you both immensely and am so proud of you.

Finally, I could not have dreamt of a partner as compassionate and understanding as Nandini. Graduate life can be stressful and frustrating at times, and I cannot recount the innumerable occasions when Nandini has allayed all my troubles and worries with her patient words of comfort and reason, interspersed with her funny antics. Nandini, I treasure every moment spent with you: from the long walks in the hallways of IIT Kanpur, to seeing the fireworks go off in Navy Pier on the 4th of July, to our adventures in Vermont, to being fascinated by the architectural wonders of Florence - we have many more miles to tread together. For being my confidante and best friend, for never failing to entertain me with your wit, humor, and sarcasm, for showering me with your love, and for always standing by my side, thank you.

TABLE OF CONTENTS

Page

LIST OF FIGURES

Figure Page

Figure Page

ABSTRACT

Mitra, Aritra Ph.D., Purdue University, August 2020. New Approaches to Distributed State Estimation, Inference and Learning with Extensions to Byzantine-Resilience. Major Professor: Shreyas Sundaram.

In this thesis, we focus on the problem of estimating an unknown quantity of interest, when the information required to do so is dispersed over a network of agents. In particular, each agent in the network receives sequential observations generated by the unknown quantity, and the collective goal of the network is to eventually learn this quantity by means of appropriately crafted information diffusion rules. The abstraction described above can be used to model a variety of problems ranging from environmental monitoring of a dynamical process using autonomous robot teams, to statistical inference using a network of processors, to social learning in groups of individuals. The limited information content of each agent, coupled with dynamically changing networks, the possibility of adversarial attacks, and constraints imposed by the communication channels, introduce various unique challenges in addressing such problems. We contribute towards systematically resolving some of these challenges.

In the first part of this thesis, we focus on tracking the state of a dynamical process, and develop a distributed observer for the most general class of LTI systems, linear measurement models, and time-invariant graphs. To do so, we introduce the notion of a multi-sensor observable decomposition - a generalization of the Kalman observable canonical decomposition for a single sensor. We then consider a scenario where certain agents in the network are compromised based on the classical Byzantine adversary model. For this worst-case adversarial setting, we identify certain fundamental necessary conditions that are a blend of system- and network-theoretic requirements. We then develop an attack-resilient, provably-correct, fully distributed state estimation algorithm. Finally, by drawing connections to the concept of age-of-information for characterizing information freshness, we show how our framework can be extended to handle a broad class of time-varying graphs. Notably, in each of the cases above, our proposed algorithms guarantee exponential convergence at any desired convergence rate.

In the second part of the thesis, we turn our attention to the problem of distributed hypothesis testing/inference, where each agent receives a stream of stochastic signals generated by an unknown static state that belongs to a finite set of hypotheses. To enable each agent to uniquely identify the true state, we develop a novel distributed learning rule that employs a min-protocol for data-aggregation, as opposed to the large body of existing techniques that rely on "belief-averaging". We establish consistency of our rule under minimal requirements on the observation model and the network structure, and prove that it guarantees exponentially fast convergence to the truth with probability 1. Most importantly, we establish that the learning rate of our algorithm is network-independent, and a strict improvement over all existing approaches. We also develop a simple variant of our learning algorithm that can account for misbehaving agents. As the final contribution of this work, we develop communication-efficient rules for distributed hypothesis testing. Specifically, we draw on ideas from event-triggered control to reduce the number of communication rounds, and employ an adaptive quantization scheme that guarantees exponentially fast learning almost surely, even when just 1 bit is used to encode each hypothesis.

1. INTRODUCTION

In various social and engineered settings, we are often faced with the problem of extracting meaningful information regarding an unknown (static or dynamic) parameter of interest, based on sequential observations of data. For instance, the unknown parameter may represent the state of a dynamical process, in which case, the data would represent measurements acquired by sensors. In a Federated learning setting, the parameter could be the best global model that is statistically consistent with the data available to a group of clients. Alternatively, the problem of interest may involve a group of individuals attempting to discern the truth regarding a topic of social or political interest, based on their own private observations (for instance, in the form of social media), and the information acquired from peers in their community. With social and engineering systems growing in size and complexity, the information needed to solve the above problems is rarely concentrated at a single point of the system. Instead, for such large-scale complex systems, information is usually dispersed across a network, and individual agents[1] in the network are endowed with signals that are only *partially informative* of the unknown parameter. In other words, each agent only has an incomplete view of the truth - a feature that underlies various inference problems over networks, and leads to the following fundamental question.

How should information be disseminated and aggregated across the network so as to enable each individual agent to eventually become knowledgeable about the unknown quantity of interest?

Several related questions naturally arise.

[1] Throughout the thesis, we will use the terms “agents” and “nodes” interchangeably.

– What are the minimal requirements on the information structure of the agents and the underlying communication network topology that facilitate inference of the unknown quantity?

– If inference is indeed possible, how fast do the agents learn the true parameter, and what factors influence the convergence rate?

– How do answers to the above questions get altered when there are temporal variations in the communication pattern of the agents?

– If certain agents misbehave or deviate from the standard protocol, under what conditions (if any) can one still hope to solve the problem at hand?

Answers to such questions can not only improve our understanding of modern networked engineering systems, but also provide critical insights into various social phenomenon. Despite extensive research on related problems over the past couple of decades, the existing body of literature remains silent on some of the questions posed above, or at best, provides partial answers. This motivates our present work.

In this thesis, we systematically develop and analyze novel algorithms (with various advantages over those already existing) to address two broad classes of problems that lie at the intersection of control theory, statistical signal processing, and network science: distributed state estimation of an LTI system, and distributed hypothesis testing (also known as distributed detection/inference/non-Bayesian social learning). In what follows, we provide a brief overview of each of these problems, and then delve into our specific contributions.

1.1 Overview and Contributions

1.1.1 Distributed State Estimation of an LTI System

Given a discrete-time LTI system $\mathbf{x}[k+1] = \mathbf{A}\mathbf{x}[k]$, and a linear measurement model $\mathbf{y}[k] = \mathbf{C}\mathbf{x}[k]$, a classical result in control theory states that one can design an observer that generates an asymptotically correct estimate $\hat{\mathbf{x}}[k]$ of the state $\mathbf{x}[k]$, if and only if the pair $(\mathbf{A}, \mathbf{C})$ is detectable. Additionally, if the pair $(\mathbf{A}, \mathbf{C})$ is observable, then one can achieve exponential convergence at any desired convergence rate. Over the last couple of decades, significant effort has been directed towards studying the distributed counterpart of the above problem, wherein observations of the process are distributed among a set of sensors modeled as nodes of a communication graph. As we describe in Chapter 2, much of the earlier work on this problem makes restrictive assumptions either on the dynamical system model, or on the observation model of the nodes, and/or the structure of the communication network. The nature of such assumptions range from restrictions on the spectrum of the state transition matrix $\mathbf{A}$, to that of local observability at each node (implying that each node can observe the entire state dynamics), to that of all-to-all communication networks (implying that each node can communicate with every other node directly). To the best of our knowledge, the authors in [1] were the first to characterize necessary and sufficient conditions for the problem under consideration. Their approach, however, requires certain nodes in the network to maintain observers of dimension greater than that of the state. Thus, at this stage, the following basic question was still left open.

Is it possible to come up with an approach that works under the same minimal assumptions as in [1], and yet requires each node to maintain an observer of dimension no more than that of the state?

In Chapter 2, we answer this question in the affirmative. While these developments are of theoretical appeal, they do not directly pave the way for practical implementations, as we discuss below.

At the heart of several applications in environmental monitoring, surveillance and patrolling, lies a distributed state estimation problem [2–13]. In such settings, a network of mobile agents are required to collectively gain information regarding the state of dynamical process evolving over a region. However, achieving such an objective in practice is fraught with various challenges, including intermittent observations of the dynamical process, loss of communication links due to mobility and packet drops, and the potential for malicious or faulty behavior by some of the agents. In particular, for mission-critical applications, adversarial attacks on certain agents can have far-reaching consequences. A specific example of such a scenario involves the use of autonomous mobile robots for estimating radiation concentrations around nuclear plants, following leakages that are either accidental or due to malicious intent [14–16]. In the first part of this thesis, we take a significant step towards addressing the challenges described above through the following sequence of developments.

- **Distributed State and Functional Observers for LTI Systems**: In Chapter 2, we develop a novel approach towards designing distributed observers for the most general class of LTI systems, sensor observation matrices, and time-invariant communication network structures. As a deviation from existing techniques, our framework requires a node to employ different strategies for estimating the locally detectable and undetectable portions of the state (w.r.t. its own measurements). Specifically, it employs consensus for estimating only the locally undetectable portion of the state, while the locally detectable portion is estimated via a standard Luenberger observer. Furthermore, the consensus weights are chosen carefully in a manner that respects the information structure of the nodes. We show that our simple, intuitive approach leads to a new class of observers with several appealing features. In particular, as mentioned earlier, the dimension of the observer maintained by each node is no more than that of the state. Finally, we establish robustness of the proposed observer to a certain class of time-varying graphs induced by communication losses.

  In practice, it may very well be the case that the quantity of interest is in fact a functional of the state, rather than the state itself. Is it possible to collaboratively estimate such a functional without having to estimate the entire state? The question posed above is particularly relevant when the dimension of the functional of interest is far lower than the dimension of the state. Motivated by this consideration, we focus on designing distributed functional observers in the latter half of Chapter 2.

- **Byzantine-Resilient Distributed Observers for LTI Systems:** In Chapter 3, we study the problem of collaborative state estimation when a certain fraction of the nodes are compromised by an adversary. As we explain in the chapter, the limited existing literature on this topic either imposes restrictive assumptions on the attack model, or provides no theoretical guarantees. In contrast, to account for worst-case adversarial behaviour, we employ the classical Byzantine adversary model where a compromised node possesses complete knowledge of the system dynamics and the network, and can deviate arbitrarily from the rules of any prescribed algorithm. We first characterize certain fundamental limitations of any distributed state estimation algorithm in terms of the measurement and communication structure of the nodes. We argue that such a characterization generalizes the conditions for non-resilient distributed state estimation on the one hand, and resilient centralized state estimation on the other. We then develop an attack-resilient, provably correct state estimation algorithm that admits a fully distributed implementation. To characterize feasible network topologies that guarantee applicability of our proposed technique, we introduce a notion of 'strong-robustness' that captures both measurement and communication redundancy. Finally, by drawing connections to bootstrap percolation theory, we argue that given an LTI system and an associated sensor network, the 'strong-robustness' property can be checked in polynomial time.

- **Distributed State Estimation over Time-Varying Graphs:** In Chapter 4, we turn our attention to dealing with the issue of time-varying communication graphs. This problem is particularly challenging since it requires analyzing the stability of a linear time-varying system where the state transition matrix at each time-step can be potentially unstable, owing to instability in the external dynamics to be tracked. The latter feature is unique to our problem of interest, and sets it apart from other distributed problems (such as consensus, optimization, or linear-equation solving) on time-varying networks where one is primarily interested in analyzing convergence properties of products of row-stochastic matrices (modulo additional perturbations that are specific to the problem). Owing to such complications, the limited literature on this topic either makes restrictive assumptions on the sequence of communication graphs (as we do in Chapter 2), or resorts to a two time-scale approach (as in [17]).[2] The main contribution of the chapter is to develop a novel single time-scale distributed observer that works under remarkably mild assumptions on the sequence of time-varying graphs in comparison with existing methods. In particular, as a key departure from existing literature, our approach can even handle scenarios with growing inter-communication intervals. Under suitable assumptions on the rate of growth of such intervals, we establish that joint observability[3] is sufficient to track the state of any discrete-time LTI system exponentially fast, at any desired rate. We also argue that via an appropriate design of the observer gains, one can achieve finite-time convergence. The main idea behind our approach is the introduction of a metric called the "freshness-index" that dynamically keeps track of the quality[4] of information being diffused across the network, and in turn, helps achieve stability of the error dynamics. Finally, we note that as the approach in Chapter 4 generalizes that of Chapter 2, convergence rate results in the former have direct implications for those in the latter. In summary, the developments in Chapters 2 and 4 indicate that a distributed observer can be appropriately designed to match (almost all) the properties of a centralized Luenberger observer - a result hitherto missing in the literature.

1.1.2 Distributed Hypothesis Testing/Inference

Consider a scenario where in a group of agents, each agent sequentially observes certain private stochastic signals over time. These signals are statistically generated

[2] Here, by a two time-scale approach, we refer to an approach that requires multiple consensus iterations between two consecutive time-steps of the dynamics.

[3] By joint observability, we imply that the state is observable w.r.t. the collective measurements of all the nodes in the network.

[4] The notion of quality that we refer to here will be made precise later in Chapter 4.

by an underlying ground truth that belongs to a finite set of hypotheses. The collective goal of the group is to infer the true state of the world that explains the joint observation profiles of the group. Of course, it may not be possible for any single agent to perform such a task in isolation owing to the *partially informative* nature of its own private signals. Such a mathematical abstraction can be used to model various social phenomenon such as the spreading of ideas, opinions on topics of social and political interest, and day-to-day decision making problems (who to vote for, which product to buy etc.) [18–22]. One can also employ such a framework to study hypothesis testing problems pertaining to engineering applications (such as object classification based on images captured by different sensors), or analyze statistical inference problems with data distributed over multiple processors [23]. In a large network, it is rarely the case that any given agent has access to the private signals of all other agents. Thus, in general, it is not possible for any given agent to employ a fully Bayesian estimator. To work around this difficulty, the existing literature adopts a belief update mechanism that requires an agent to combine a local Bayesian update (based on private observations) with a consensus-based opinion pooling of neighboring beliefs. In particular, such consensus-based pooling refers to either a weighted arithmetic average of beliefs, or a weighted geometric average. In this context, our contributions are as follows.

- **A Novel Distributed Learning Rule: Improved Learning Rate and Extension to Byzantine-Resilience:** In Chapter 5, we propose a belief update rule that differs fundamentally from those in the existing literature. Specifically, our learning rule does not employ any form of "belief-averaging" or linear consensus based belief aggregation protocol. Instead, our approach requires each agent to maintain two belief vectors, namely, a local belief vector and an actual belief vector. For each hypothesis, an agent updates its local belief on that hypothesis based on standard Bayes rule. This is done without any network influence. Subsequently, the agent updates its actual belief on that hypothesis (up to normalization) as the minimum of its own local belief (on that hypothesis) and the actual beliefs of its neighbors. We show that under *minimal* requirements on the information structures of the agents and the underlying communication graph, every agent is able to learn the true state of the world asymptotically almost surely. Given the nonlinear nature of our proposed update rule, existing analysis techniques that study linear belief recursions cannot be adapted to suit our needs. This requires us to develop a novel sample path based proof technique to establish consistency of our learning rule.

As the most significant contribution of this chapter, we prove that with probability 1, every agent rules out each false hypothesis exponentially fast, at a *network-independent* rate that *strictly improves* upon all existing learning rates. Specifically, with existing "belief-averaging" schemes, the asymptotic rate at which a particular false hypothesis is ruled out is given by a convex combination of the agents' relative entropies, where the convex weights are the eigenvector centralities of the agents. In contrast, the corresponding rate for our rule is the *best* relative entropy (between the true state and the false hypothesis under consideration) across the network. Unlike existing approaches, such a rate does not exhibit any dependence on the size or structure of the network, and is solely determined by the observation model of the agents. A key implication of this fact is that the long-run performance of our rule does not get impacted by the specific manner in which signal structures are allocated across the network, as long as the total information content remains the same.

The final contribution of Chapter 5 is to propose and analyze a simple and computationally-efficient variant of our learning rule that guarantees exponentially fast learning with probability 1, despite the presence of misbehaving agents modeled as per the classical Byzantine adversary model.[5]

- **Distributed Hypothesis Testing with Sparse and Quantized Communication:** In Chapter 6, we investigate the distributed hypothesis testing prob-

[5] Given the context of social learning, such agents can be used model stubborn individuals or religious, political extremists.

lem when communication between agents is costly, and takes place over channels with finite bandwidth. Our focus is on answering the following basic questions. (i) How frequently should the agents interact to learn the truth? (ii) What piece of information needs to be exchanged? (iii) How much can we compress the information being transmitted? Addressing these questions is critical to resolving the *communication bottleneck* in modern distributed systems where devices run on low-battery power, setting up communication links incur great latency, and channels can support only a finite number of bits. To reduce the number of communication rounds, we propose an event-triggered distributed learning algorithm where an agent broadcasts only those components of its belief vector that have adequate innovation, to only those neighbors that are in need of such information. We prove that incorporating feedback from neighbors enables our rule to significantly reduce information flow from uninformative agents to informative agents. We identify sparse communication regimes where the inter-communication intervals between the agents grow unbounded over time, and yet, all agents end up learning the truth exponentially fast almost surely at the best known rate for this problem (which is precisely the rate that we obtain in Chapter 5.) We also discuss various trade-offs between the rate of convergence and the sparsity of the communication pattern.

We next ask: To learn the true state, how many bits must an agent use to encode its belief on each hypothesis? To answer this question, we develop a novel algorithm that leverages the idea of adaptive quantization. We show that, if each agent sequentially refines the range of its quantizers, then all agents can learn the truth exponentially fast under our rule, while using just 1 bit to encode each hypothesis. Our findings reveal a relationship between the learning rate and the quantizer precision levels. Using this result, we show that if the number of bits used to encode each hypothesis is chosen to be large enough, then one can recover the exact same learning rate as with infinite precision. In sum, we develop communication-efficient distributed learning algorithms that provide strong theoretical guarantees in the face of sparse and imprecise communication.

1.2 Notation and Terminology

A directed graph is denoted by $\mathcal{G} = (\mathcal{V}, \mathcal{E})$, where $\mathcal{V} = \{1, \cdots, N\}$ is the set of nodes and $\mathcal{E} \subseteq \mathcal{V} \times \mathcal{V}$ represents the edges. An edge from node j to node i, denoted by (j, i), implies that node j can transmit information to node i. The neighborhood of the i-th node is defined as $\mathcal{N}_i \triangleq \{i\} \cup \{j \,|\, (j, i) \in \mathcal{E}\}$. A node j is said to be an out-neighbor of node i if $(i, j) \in \mathcal{E}$. By an *induced* subgraph of $\mathcal{G}$ obtained by removing certain nodes $\mathcal{C} \subset \mathcal{V}$, we refer to the subgraph that has $\mathcal{V} \setminus \mathcal{C}$ as its node set and contains only those edges of $\mathcal{E}$ with both end points in $\mathcal{V} \setminus \mathcal{C}$. The notation $|\mathcal{V}|$ is used to denote the cardinality of a set $\mathcal{V}$. Throughout this thesis, the terms ‘network’ and ‘communication graph’ are used interchangeably.

The set of all eigenvalues of a matrix $\mathbf{A}$ is denoted by $sp(\mathbf{A}) \triangleq \{\lambda \in \mathbb{C} \,|\, det(\mathbf{A} - \lambda\mathbf{I}) = 0\}$. The set of all marginally stable and unstable eigenvalues of a matrix $\mathbf{A}$ is denoted by $\Lambda_U(\mathbf{A}) \triangleq \{\lambda \in sp(\mathbf{A}) \,|\, |\lambda| \geq 1\}$. For a matrix $\mathbf{A}$, we use $a_{\mathbf{A}}(\lambda)$ and $g_{\mathbf{A}}(\lambda)$ to denote the algebraic and geometric multiplicities, respectively, of an eigenvalue $\lambda \in sp(\mathbf{A})$. An eigenvalue λ is said to be simple if $a_{\mathbf{A}}(\lambda) = g_{\mathbf{A}}(\lambda) = 1$. We use $\mathcal{R}(\mathbf{A})$ to denote the row-space of $\mathbf{A}$ and $\mathbf{A}^{\dagger}$ to refer to its Moore Penrose inverse. Unless otherwise specified, we will use $\|\mathbf{A}\|$ to represent the induced 2-norm of a matrix $\mathbf{A}$. For a set $\{\mathbf{A}_1, \cdots, \mathbf{A}_n\}$ of matrices, we use the notation $diag(\mathbf{A}_1, \cdots, \mathbf{A}_n)$ to refer to a block diagonal matrix with the matrix $\mathbf{A}_i$ as the i-th block-diagonal entry. For a set $\mathcal{S} = \{s_1, \cdots, s_p\} \subseteq \{1, \cdots, N\}$, and a matrix $\mathbf{C} = \begin{bmatrix} \mathbf{C}_1^T & \cdots & \mathbf{C}_N^T \end{bmatrix}^T$, we define $\mathbf{C}_{\mathcal{S}} \triangleq \begin{bmatrix} \mathbf{C}_{s_1}^T & \cdots & \mathbf{C}_{s_p}^T \end{bmatrix}^T$. We use the star notation to avoid writing matrices that are either unimportant or that can be inferred from context. We use $\mathbf{1}_n$ to denote a column vector of dimension n that has all its components equal to 1, and $\mathbf{I}_r$ to denote an identity matrix of dimension $r \times r$. We use $\mathbb{N}$ and $\mathbb{N}_+$ to refer to the set of non-negative integers and the set of positive integers, respectively.

1.3 Thesis Outline

In Chapter 2, we develop a distributed state estimation framework for the most general class of LTI systems, sensor observation matrices, and time-invariant communication graphs. In Chapter 3, we focus on developing provably correct, fully distributed state estimation algorithms that are immune to worst-case adversarial attacks on certain nodes of the network. In Chapter 4, we extend our distributed observer framework to account for a broad class of time-varying communication graphs. In Chapter 5, we develop a fundamentally novel distributed learning rule, establish its consistency under minimal assumptions, prove that it leads to the fastest convergence rate in the existing literature on the topic, and show that it can be easily and efficiently extended to account for misbehaving entities in the network. In Chapter 6, we develop communication-efficient distributed inference algorithms that guarantee exponential convergence despite quantized communication, and a significant reduction in the number of communication rounds. In Chapter 7, we summarize the main findings of this thesis, and identify several interesting future directions.

This work was supported in part by NSF CAREER award 1653648, and by the Laboratory Directed Research and Development program at Sandia National Laboratories.

Part I

Distributed State Estimation of LTI Systems

2. DISTRIBUTED STATE AND FUNCTIONAL OBSERVERS FOR LTI SYSTEMS

In this chapter, we first consider the problem of distributed state estimation of a linear time-invariant (LTI) system by a network of sensors. We develop a distributed observer that guarantees asymptotic reconstruction of the state for the most general class of LTI systems, sensor network topologies and sensor measurement structures. Our analysis builds upon the following key observation - a given node can reconstruct a portion of the state solely by using its own measurements and constructing appropriate Luenberger observers; hence, it only needs to exchange information with neighbors (via consensus dynamics) for estimating the portion of the state that is not locally detectable. This intuitive approach leads to a new class of distributed observers with several appealing features. Furthermore, by imposing additional constraints on the system dynamics and network topology, we show that it is possible to construct a simpler version of the proposed distributed observer that achieves the same objective while admitting a fully distributed design phase. We then establish the robustness of our general framework to a certain class of time-varying graphs that are a consequence of communication losses.

In the latter half of the chapter, we focus on the problem of collaborative estimation of certain functionals of the state. We first show that classical existence conditions for the design of centralized functional observers do not directly translate to the distributed setting, due to the coupling that exists between the dynamics of the functionals of interest and the diverse measurements at the various nodes. Accordingly, we design transformations that reveal such couplings and identify portions of the corresponding dynamics that are locally detectable at each sensor node. We then leverage our distributed observer design framework to allow each node to asymptotically estimate the desired functionals, and identify sufficient conditions for this to happen.

2.1 Introduction

In many applications involving large-scale complex systems (such as the power grid, transportation systems, industrial plants, etc.), the state of the system is monitored by a group of sensors spatially distributed over large sparse networks where the communication between sensors is limited (see [24, 25]). To model such a scenario, consider the discrete-time linear time-invariant dynamical system[1]

$$\mathbf{x}[k+1] = \mathbf{A}\mathbf{x}[k], \tag{2.1}$$

where $k \in \mathbb{N}$ is the discrete-time index, $\mathbf{x}[k] \in \mathbb{R}^n$ is the state vector, and $\mathbf{A} \in \mathbb{R}^{n \times n}$ is the system matrix. The state of the system is monitored by a network of N sensors, each of which receives a partial measurement of the state at every time-step. Specifically, the i-th sensor has access to a measurement of the state, given by

$$\mathbf{y}_i[k] = \mathbf{C}_i\mathbf{x}[k], \tag{2.2}$$

where $\mathbf{y}_i[k] \in \mathbb{R}^{r_i}$ and $\mathbf{C}_i \in \mathbb{R}^{r_i \times n}$. We use $\mathbf{y}[k] = \begin{bmatrix}\mathbf{y}_1^T[k] & \cdots & \mathbf{y}_N^T[k]\end{bmatrix}^T$ to represent the collective measurement vector, and $\mathbf{C} = \begin{bmatrix}\mathbf{C}_1^T & \cdots & \mathbf{C}_N^T\end{bmatrix}^T$ to denote the collection of the sensor observation matrices. These sensors are represented as nodes of an underlying directed communication graph which governs the information flow between the sensors.

Each node is capable of exchanging information with its neighbors and performing computational tasks. The goal of each node is to estimate the entire system state $\mathbf{x}[k]$ based on its respective (limited) state measurements and the information obtained from neighbors. This is known as the *distributed state estimation problem.*

[1] Although we consider noiseless dynamics for clarity of exposition (like [1, 26–32]), the techniques developed in this chapter guarantee bounded mean square estimation error in the presence of i.i.d. process and measurement noise with bounded second moments.

In this chapter, our primary **objective** is to design a distributed algorithm that guarantees asymptotic reconstruction of the entire state $\mathbf{x}[k]$ at each node. The problem we study is formally stated in Section 2.2. For much of the chapter, we will focus on developing theory for *linear time-invariant systems* and *time-invariant directed communication graphs.* In Section 2.7, however, we shall establish that our proposed framework can be extended to account for certain types of time-varying networks that may arise as a consequence of intermittent communication link failures.

The chapter is organized as follows. In Section 2.2, we formally describe the distributed state estimation problem, discuss related work and summarize our contributions. Some preliminary ideas and terminology required for subsequent analysis are presented in Section 2.3. Section 2.4 highlights the key ideas of our distributed estimation scheme via a simple illustrative example. In Section 2.5, we solve the most general version of the problem, whereas in Section 2.6 we provide a solution strategy for a simpler variant of the original problem that enjoys several implementation benefits. We extend our approach to account for a certain class of time-varying graphs in Section 2.7. In Section 2.9, we generalize our results to the scenario where the nodes are interested in estimating certain functionals of the state.

2.2 Problem Formulation

Consider the LTI system given by (2.1), the measurement model specified by (2.2), and a predefined directed communication graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$, where $\mathcal{V}$ represents the set of N nodes (or sensors). Each node i maintains an estimate $\hat{\mathbf{x}}_i[k]$ of the state $\mathbf{x}[k]$ of system (2.1), and updates such an estimate based on information received from its neighbors and its local measurements (if any). To formally define the problem under study, we use the following terminology.

Definition 2.2.1 (Distributed Observer) *A set of state estimate update and information exchange rules is called a distributed observer if* $\lim_{k\to\infty} \|\hat{\mathbf{x}}_i[k] - \mathbf{x}[k]\| =$

$0, \forall i \in \{1, \cdots, N\}$, *i.e., the state estimate maintained by each node asymptotically converges to the true state of the plant.*

There are various technical challenges associated with constructing a distributed observer. First, if the pair $(\mathbf{A}, \mathbf{C}_i)$ is not detectable for some (or all) $i \in \{1, \cdots, N\}$, then the corresponding nodes cannot estimate the true state of the plant based on their own local measurements, thereby dictating the need to exchange information with other nodes. Second, this exchange of information is restricted by the underlying communication graph $\mathcal{G}$. With these challenges in mind, we address and solve the following problem in this chapter.

Problem 1 *Design a distributed observer for LTI systems of the form* (2.1)*, linear measurement models of the form* (2.2)*, and time-invariant directed communication graphs.*

There are a variety of approaches to construct distributed observers (as defined in Definition 2.2.1) that have been proposed in the literature, which we will now review. After that, we will summarize how our approach differs from the existing approaches, before delving into the details of our construction.

2.2.1 Related Work

The papers [33–35] consider distributed estimation of scalar stochastic dynamical systems over general graphs; in these works, it is typically assumed that each node receives scalar local observations, leading to local observability at every node. The papers [36, 37] consider a version of this problem where the underlying communication graph is assumed to be complete. For more general stochastic systems, the Kalman filtering based approach to solving the distributed estimation problem has been explored by several researchers. The approach proposed in [38–40] relies on a two-step strategy - a Kalman filter based state estimate update rule, and a data fusion step based on average-consensus. The stability and performance issues of this method have been investigated in [41, 42]. A drawback of this method (and the ones in [34, 43, 44]), stems from the fact that they require a (theoretically) infinite number of data fusion iterations between two consecutive time steps of the plant dynamics in order to reach average consensus, thereby leading to a two-time-scale algorithm. More recently, finite-time data fusion has been studied in [45] and [46]. Although an improvement over the infinite-time data fusion case, these methods still rely on a two-time-scale strategy. Single-time-scale distributed filtering techniques are proposed in [47–53]. These approaches involve LMI-based feasibility conditions (the method in [52] requires satisfaction of certain nonlinear matrix inequalities) and in general do not shed light on the network conditions required to satisfy such LMIs.

In [26], [27], and [31], sufficient conditions are presented for a distributed observer to exist in undirected networks. Specifically, in [26] and [27], the authors propose a *scalar-gain estimator* that runs on a single-time-scale.[2] They introduce the notion of "Network Tracking Capacity" (NTC), a measure of the most unstable dynamics (in terms of the 2-norm of the state matrix) that can be estimated with bounded mean-squared error under their scheme. However, the tight coupling between the network and the plant dynamics typically limits the set of unstable eigenvalues that can be accommodated by their method without violating the constraints imposed upon the range of the scalar gain parameter. In [32, 54], the author approaches the observer design problem from a geometric perspective and provides separate necessary and sufficient conditions for consensus-based distributed observer design. In [1, 28–30], the authors use single-time-scale algorithms, and work under the broadest assumptions, namely that the pair $(\mathbf{A}, \mathbf{C})$ is detectable, where $\mathbf{A}$ represents the system matrix, and $\mathbf{C}$ is the collection of all the node observation matrices. In all of these works, the authors rely on state augmentation[3] for casting the distributed estimation problem

[2] By a single-time-scale algorithm, we imply an algorithm where each node operates at the same time-scale as the plant, and updates its estimate and transmits information to neighbors only once in each time-step.

[3] In these works, some nodes maintain observers of dimension larger than that of the state of the plant; hence, such observers are referred to as augmented observers, and the state they estimate is referred to as an augmented state.

as a problem of designing a decentralized stabilizing controller for an LTI plant, using the notion of fixed modes [55,56]. Specifically, in [1], the authors relate the distributed observer design problem for directed networks to the detectability of certain strongly connected clusters within the network, and provide a single necessary and sufficient condition for their scheme.

2.2.2 Summary of Contributions

In this chapter, we provide a new approach to designing distributed observers for LTI dynamical systems. Specifically, we use the following simple, yet key observation - for each node, there may be certain portions of the state that the node can reconstruct using only its local measurements. The node thus does so. For the remaining portion of the state space, the node relies on a consensus-based update rule. The key is that those nodes that can reconstruct certain states on their own act as "*root nodes*" (or "*leaders*") in the consensus dynamics, leading the rest of the nodes to asymptotically estimate those states as well. These ideas, in a nutshell, constitute the essence of our distributed estimation strategy.

We begin by considering the most general category of systems and graphs (taken together) for which a distributed observer can be constructed, and develop an estimation scheme that enjoys the following appealing features *simultaneously*, thereby differentiating our work from the existing literature discussed in Section 2.2.1.

(i) It provides theoretical guarantees regarding the design of asymptotically stable estimators.

(ii) It results in a single-time-scale algorithm.

(iii) It does not require any state augmentation.

(iv) It requires only state estimates to be exchanged locally.

(v) It works under the broadest conditions on the system and communication graph.

Subsequently, for a certain subclass of systems and communication graphs, we provide a simpler fully distributed estimation scheme (at both design- and run-time) for achieving asymptotic state reconstruction. We show that our proposed framework can be extended to guarantee asymptotic state reconstruction in the presence of communication losses that lead to time-varying networks. Finally, we generalize our results to the case where the nodes seek to collaboratively estimate certain functionals of the state.

The results presented in this chapter were published as [57–59].

2.3 Preliminaries

Before we proceed with a formal analysis of the problem of designing a distributed observer, we first identify the main consideration that shall dictate our solution strategy, namely, *the relationship between the measurement structure of the nodes and the underlying communication graph.* To classify sets of systems and graphs based on this relationship, we need to first establish some notation. Accordingly, for each node i, we denote the detectable and undetectable eigenvalues[4] of $\mathbf{A}$ by the sets $\mathcal{O}_i$ and $\mathcal{UO}_i$, respectively. We define $\sigma_i \triangleq |\mathcal{O}_i|$. Next, we introduce the notion of *root nodes.*

Definition 2.3.1 (Root nodes) *For each $\lambda_j \in \Lambda_U(\mathbf{A})$, the set of nodes that can detect λ_j is denoted by $\mathcal{S}_j$, and called the set of root nodes for λ_j.*

We also recall the definition of a source component of a graph [1].

Definition 2.3.2 (Source Component) *Given a directed graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$, a source component $(\mathcal{V}_s, \mathcal{E}_s)$ is defined as a strongly connected component of $\mathcal{G}$ such that there are no edges from $\mathcal{V} \setminus \mathcal{V}_s$ to $\mathcal{V}_s$.*

Let there be p source components of $\mathcal{G}$, denoted by $\{(\mathcal{V}_i, \mathcal{E}_i)\}_{i\in\{1,\cdots,p\}}$. The subsystem associated with the i-th source component is given by the pair $(\mathbf{A}, \mathbf{C}_{\mathcal{V}_i})$. For the

[4] Throughout the chapter, for the sake of conciseness, we use the terminology 'node i can detect eigenvalue λ_j' to imply that $rank \begin{bmatrix} \mathbf{A}-\lambda_j\mathbf{I}_n \\ \mathbf{C}_i \end{bmatrix} = n$. Each stable eigenvalue of $\mathbf{A}$ is by default considered to be detectable w.r.t. the measurements of each node.

subsequent development, it should be noted that by a system $(\mathbf{A}, \mathbf{C})$, we refer to the matrix $\mathbf{A}$ in equation (2.1), and the matrix $\mathbf{C} = \begin{bmatrix} \mathbf{C}_1^T & \cdots & \mathbf{C}_N^T \end{bmatrix}^T$ containing each of the measurement matrices given by (2.2). Then, we classify systems and graphs based on the following two conditions.

Condition 1 *A system $(\mathbf{A}, \mathbf{C})$ and graph $\mathcal{G}$ are said to satisfy Condition 1 if the subsystem associated with every source component is detectable, i.e., the pair $(\mathbf{A}, \mathbf{C}_{\mathcal{V}_i})$ is detectable $\forall i \in \{1, \cdots, p\}$.*

Condition 2 *A system $(\mathbf{A}, \mathbf{C})$ and graph $\mathcal{G}$ are said to satisfy Condition 2 if for each unstable or marginally stable eigenvalue of the plant, there exists at least one root node within each source component, i.e., for all $i \in \{1, \cdots, p\}$ and all $\lambda_j \in \Lambda_U(\mathbf{A})$, there exists $l \in \mathcal{V}_i$, such that rank $\begin{bmatrix} \mathbf{A} - \lambda_j \mathbf{I}_n \\ \mathbf{C}_l \end{bmatrix} = n$.*

Note that given a source component, Condition 2 does not necessarily imply the existence of a single node within such a component that can simultaneously detect all the unstable and marginally stable eigenvalues of the system via its own measurements.

Remark 2.3.1 *It is trivial to see that if a system $(\mathbf{A}, \mathbf{C})$ and graph $\mathcal{G}$ satisfy Condition 2, they also satisfy Condition 1. To see that the converse is not true in general, consider the 3-node network $\mathcal{G}$ in Figure 2.1, and the following model:*

$$\mathbf{A} = \begin{bmatrix} 2 & 0 \\ 0 & 2 \end{bmatrix}, \mathbf{C}_1 = \begin{bmatrix} 1 & 0 \end{bmatrix}, \mathbf{C}_2 = \begin{bmatrix} 0 & 1 \end{bmatrix}, \mathbf{C}_3 = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}. \tag{2.3}$$

From Figure 2.1, we see that the network has two source components, namely, the strong component formed by nodes 1 and 2 (S_1), and the isolated node 3 (S_2). Clearly, each of the pairs $(\mathbf{A}, \mathbf{C}_3)$ and $(\mathbf{A}, \begin{bmatrix} \mathbf{C}_1 \\ \mathbf{C}_2 \end{bmatrix})$ are detectable. Thus, the system is detectable from each of the two source components. It follows that this system and graph satisfy Condition 1. However, neither node 1 nor node 2 can detect the eigenvalue $\lambda = 2$

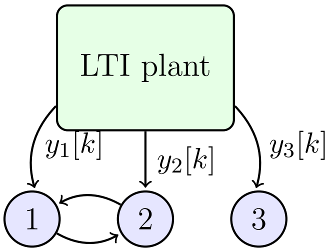


Fig. 2.1. Example for illustrating Remark 2.3.1.

based on just their own measurements, i.e., there does not exist a root node for $\lambda = 2$ *within source component* S_1. *Thus, this system and graph do not satisfy Condition 2.*

In [1], the authors identified that a distributed observer cannot be constructed (regardless of the state update or exchange rules) if the system $(\mathbf{A}, \mathbf{C})$ and graph $\mathcal{G}$ do not satisfy Condition 1. They then designed a distributed observer for the class of systems and graphs satisfying Condition 1 by constructing augmented state observers (i.e., observers of dimension larger than that of the system) drawing upon connections to decentralized control theory. Here, we present an alternate and more direct design approach, and in the process, establish that it is possible to design a distributed observer *without state augmentation* for this (most general) class of systems and graphs.[5] Before we delve into the specifics of the distributed observer design for systems and graphs satisfying Condition 1, we present a simple motivating example which serves to build intuition for the more complicated scenarios.[6]

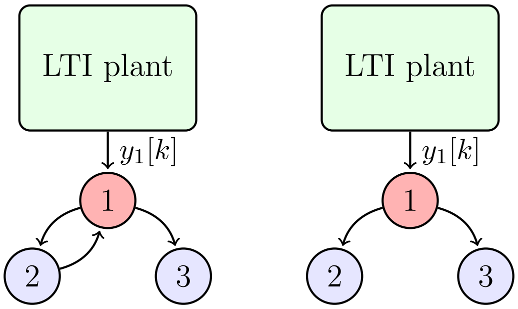


Fig. 2.2. The graph on the left is the actual network. The graph on the right is a DAG constructed from the original graph.

2.4 Illustrative Example

Consider a scalar unstable plant with dynamics given by $x[k+1] = 1.5x[k]$. The plant is monitored by a network of nodes, as depicted by Figure 2.2. Node 1 has a measurement given by $y_1[k] = x[k]$, whereas nodes 2 and 3 have no measurements. Given this plant and network model, we wish to design a distributed observer. The commonly adopted approach in the literature is to develop a consensus-based state estimate update rule for each node in the network [1, 26–29]. Here, we make the following observation: since node 1 can detect the eigenvalue $\lambda = 1.5$ of the plant based on its own measurements, it can run a Luenberger observer for estimating $x[k]$, without requiring information from its neighbors. Specifically, the following Luenberger observer allows node 1 to estimate and predict the state:

$$\hat{x}_1[k+1] = 1.5\hat{x}_1[k] + 1.5(y_1[k] - \hat{x}_1[k]) = 1.5y_1[k]. \tag{2.4}$$

[5] The exact structure of our distributed observer presented in Section 2.5.5 illustrates that the dimension of the internal state/estimate $\hat{\mathbf{x}}_i[k]$ maintained by a given node i is equal to the dimension of the state $\mathbf{x}[k]$.

[6] At this point, it is worth mentioning that although the distributed observer that we shall design for systems and graphs satisfying Condition 1 will also work for systems and graphs satisfying Condition 2, we will later propose an alternate scheme with various implementation benefits for the latter class of systems and graphs.

Here, $\hat{x}_1[k]$ is the estimate of $x[k]$ maintained by node 1 at time-step k. Now suppose nodes 2 and 3 update their respective estimates of $x[k]$ as follows: $\hat{x}_i[k+1] = 1.5\hat{x}_1[k], i = 2, 3$. Since $\lim_{k\to\infty} |\hat{x}_1[k] - x[k]| = 0$ based on the Luenberger observer dynamics given by (2.4), it is easy to see that the estimates of nodes 2 and 3 also converge to the true state $x[k]$. This simple example illustrates the following key observations. (i) It is not necessary for every node in the network to run consensus dynamics for estimating the state. More generally, a node needs to run consensus for estimating only the portion of the state vector that is not locally detectable. The rest of the state can be estimated via appropriately designed Luenberger observers. (ii) An inspection of the observable subspace of each node guides the decision of participating (or not participating) in consensus for the example we considered. For more general system and measurement matrices, we shall rely on appropriate similarity transformations which shall reveal what a node can or cannot observe. (iii) Although node 1 is in a position to receive information from node 2, it chooses not to listen to any of its neighbors. This pattern of information flow results in a special Directed Acyclic Graph (DAG) of the original network, rooted at node 1. In the DAG constructed in the illustrative example, node 1 can be viewed as the *source* of information for the state $x[k]$, and the DAG structure can be viewed as the medium for transmitting information from the source to the rest of the network, without corrupting the source itself (this is achieved in this example by ignoring the edge from node 2 to node 1). Under this approach, note that every node maintains an observer of dimension 1, which is equal to the dimension of the state (i.e., there is no state augmentation). Based on these observations, we are now ready to extend the ideas conveyed by this simple example for tackling more general systems and networks.[7]

[7] Notice that the original network in this illustrative example has only one source component comprised of the nodes 1 and 2, and node 1 is a root node for $\lambda = 1.5$ (node 1 can detect $\lambda = 1.5$). Thus, the system and graph illustrated in this example satisfy Condition 2, and hence also Condition 1.

2.5 Estimation Scheme for systems and graphs satisfying Condition 1

In this section, we develop a distributed observer for systems and graphs satisfying Condition 1. For presenting the key ideas while reducing notational complexity, we shall make the following assumption.

Assumption 2.5.1 *The graph $\mathcal{G}$ is strongly connected, i.e., there exists a directed path from any node i to any other node j, where $i, j \in \mathcal{V}$.*

Later, we shall argue that the development can be easily extended to any general directed network. For now, it suffices to say that any directed graph can be decomposed into strong components, some of which are source components (strong components with no incoming edges from the rest of the network); the strategy that we develop here for a strongly connected graph will be employed within each source component.

Remark 2.5.2 *Since we are focusing on systems and graphs satisfying Condition 1, it follows that under Assumption 2.5.1, the pair $(\mathbf{A}, \mathbf{C})$ is detectable (as a strongly connected graph is one single source component).*

Remark 2.5.3 *Note that under Condition* 1 *with a strongly connected graph, one might consider the possibility of aggregating all the sensor measurements at a central node and constructing a centralized Luenberger observer, leveraging the fact that $(\mathbf{A}, \mathbf{C})$ is detectable. However, for large networks, the routing of measurement information to and from such a central node via multiple hops would induce delays. A distributed approach (such as the one considered in this chapter) alleviates such a difficulty.*

We are now in a position to detail the steps to be followed for designing a distributed observer for systems and graphs satisfying Condition 1. We start by providing a generalization of the Kalman observable canonical form to a setting with multiple sensors.

$$\underbrace{\begin{bmatrix}1&2&-2&-15\\0&2&4&-16\\0&0&3&-3\\0&0&0&0\end{bmatrix}}_{\mathbf{A}} \xrightarrow{\mathbf{T}_1} \underbrace{\left[\begin{array}{c|ccc}1&&\mathbf{0}&\\ \hline -188.18&1.24&6.79&-19.04\\253.44&0.35&-0.08&6.16\\76.18&-0.04&-0.78&3.85\end{array}\right]}_{\bar{\mathbf{A}}_1} \xrightarrow{\mathbf{T}_2} \underbrace{\left[\begin{array}{c|c|cc}1&&\mathbf{0}&\\ \hline -81.42&2&\mathbf{0}&\\ \cline{3-4} 262.1&-14.86&-7.96&12.48\\84.34&-10.01&-6.99&10.96\end{array}\right]}_{\bar{\mathbf{A}}_2} \xrightarrow{\mathbf{T}_3} \underbrace{\left[\begin{array}{c|c|c|c}1&&\mathbf{0}&\\ \hline -81.42&2&\mathbf{0}&\\ \cline{2-4} 10.84&0.07&3&0\\ \cline{3-4} 266.34&-17.91&-125.33&0\end{array}\right]}_{\bar{\mathbf{A}}=\bar{\mathbf{A}}_3}$$

$$\underbrace{\begin{bmatrix}7&-14&35&14\\0&2&-8&-4\\0&0&5&-5\end{bmatrix}}_{\mathbf{C}=\left[\mathbf{C}_1^T\ \ \mathbf{C}_2^T\ \ \mathbf{C}_3^T\right]^T} \qquad \underbrace{\left[\begin{array}{c|ccc}1666&0&0&0\end{array}\right]}_{\bar{\mathbf{C}}_1} \qquad \underbrace{\left[\begin{array}{cc|cc}-364&4.47&0&0\end{array}\right]}_{\bar{\mathbf{C}}_2} \qquad \underbrace{\left[\begin{array}{c|c|c|c}1666&0&0&0\\ \hline -364&4.47&0&0\\ \hline 105&2.94&41.45&0\end{array}\right]}_{\bar{\mathbf{C}}=\left[\bar{\mathbf{C}}_1^T\ \ \bar{\mathbf{C}}_2^T\ \ \bar{\mathbf{C}}_3^T\right]^T}$$

$$\mathbf{T}_1=\begin{bmatrix}7&0.3430&-0.8575&-0.3430\\-14&0.8996&0.2511&0.1004\\35&0.2511&0.3723&-0.2511\\14&0.1004&-0.2511&0.8996\end{bmatrix},\ \mathbf{T}_2=\begin{bmatrix}1&0&0&0\\0&-0.611&-0.6964&-0.6569\\0&-1.4724&0.6238&-0.3549\\0&-1.3890&-0.3549&0.6652\end{bmatrix},\ \mathbf{T}_3=\begin{bmatrix}1&0&0&0\\0&1&0&0\\0&0&3.4616&0.8431\\0&0&-5.4281&0.5377\end{bmatrix}$$

Fig. 2.3. Illustration of the Multi-Sensor Observable Canonical Decomposition for a detectable pair $(\mathbf{A}, \mathbf{C})$.

2.5.1 Multi-Sensor Observable Canonical Decomposition

Given a system matrix $\mathbf{A}$ and a set of N sensors where the i-th sensor has an observation matrix given by $\mathbf{C}_i$, we introduce the notion of a *multi-sensor observable canonical decomposition* in this section. The basic philosophy underlying such a decomposition is as follows: given a list of indexed sensors, perform an observable canonical decomposition with respect to the first sensor. Then, identify the observable portion of the state space with respect to sensor 2 *within the unobservable subspace of sensor 1*, and repeat the process until the last sensor is reached. Thus, one needs to perform N observable canonical decompositions, one for each sensor, with the last decomposition revealing the portions of the state space that can and cannot be observed using the cumulative measurements of all the sensors. The details of the multi-sensor observable canonical decomposition are captured by the proof of the following result (given in Section 2.11.1).

Proposition 2.5.1 *Given a system matrix $\mathbf{A}$, and a set of N sensor observation matrices $\mathbf{C}_1, \mathbf{C}_2, \cdots, \mathbf{C}_N$, define $\mathbf{C} \triangleq \begin{bmatrix} \mathbf{C}_1^T & \cdots & \mathbf{C}_N^T \end{bmatrix}^T$. Then, there exists a similarity transformation matrix $\mathcal{T}$ which transforms the pair $(\mathbf{A}, \mathbf{C})$ to $(\bar{\mathbf{A}}, \bar{\mathbf{C}})$, such that*

$$\bar{\mathbf{A}} = \left[\begin{array}{c|c|c|c|c}
\mathbf{A}_{11} & \multicolumn{4}{c}{\mathbf{0}} \\ \hline
\mathbf{A}_{21} & \mathbf{A}_{22} & \multicolumn{3}{c}{\mathbf{0}} \\ \cline{2-5}
\vdots & \vdots & \ddots & \vdots & \vdots \\
\mathbf{A}_{(N-1)1} & \mathbf{A}_{(N-1)2}\cdots & \mathbf{A}_{(N-1)(N-1)} & \multicolumn{2}{c}{\mathbf{0}} \\ \cline{3-5}
\mathbf{A}_{N1} & \mathbf{A}_{N2}\ \cdots & \mathbf{A}_{N(N-1)} & \mathbf{A}_{NN} & \mathbf{0} \\ \cline{4-5}
\mathbf{A}_{1} & \mathbf{A}_{2}\ \cdots & \mathbf{A}_{(N-1)} & \mathbf{A}_{N} & \mathbf{A}_{\mathcal{U}}
\end{array}\right], \tag{2.5}$$

$$\bar{\mathbf{C}} = \begin{bmatrix} \bar{\mathbf{C}}_1 \\ \bar{\mathbf{C}}_2 \\ \vdots \\ \bar{\mathbf{C}}_N \end{bmatrix} = \left[\begin{array}{c|c|ccc}
\mathbf{C}_{11} & \multicolumn{4}{c}{\mathbf{0}} \\ \hline
\mathbf{C}_{21} & \mathbf{C}_{22} & \multicolumn{3}{c}{\mathbf{0}} \\ \hline
\vdots & \vdots & \vdots & \vdots & \vdots \\
\mathbf{C}_{N1} & \mathbf{C}_{N2} & \cdots\mathbf{C}_{N(N-1)} & \mathbf{C}_{NN} & \multicolumn{1}{|c}{\mathbf{0}}
\end{array}\right].$$

Furthermore, the following properties hold: (i) the pair $(\mathbf{A}_{ii}, \mathbf{C}_{ii})$ is observable $\forall i \in \{1, 2, \cdots, N\}$; and (ii) the matrix $\mathbf{A}_{\mathcal{U}}$ describes the dynamics of the unobservable subspace of the pair $(\mathbf{A}, \mathbf{C})$.

Figure 3 illustrates the steps of the multi-sensor observable canonical decomposition for a sensor network with 3 nodes. The first step involves an observable canonical decomposition of the pair $(\mathbf{A}, \mathbf{C}_1)$ via the matrix $\mathbf{T}_1$. Next, $\mathbf{T}_2$ reveals the portion of the unobservable subspace of $(\mathbf{A}, \mathbf{C}_1)$ that can be observed using the observation matrix $\mathbf{C}_2$. Finally, $\mathbf{T}_3$ reveals the portion of the unobservable subspace of $(\mathbf{A}, \begin{bmatrix} \mathbf{C}_1 \\ \mathbf{C}_2 \end{bmatrix})$ that can be observed using the observation matrix $\mathbf{C}_3$. For this example, we have $\mathcal{T} = \mathbf{T}_1\mathbf{T}_2\mathbf{T}_3$. In the following section, we discuss how the multi-sensor observable canonical decomposition is applicable to the problem of designing a distributed observer for systems and graphs satisfying Condition 1.

Remark 2.5.4 *Note that while describing the multi-sensor observable canonical decomposition, we did not specify any rule for indexing the sensors. This is precisely because the technique we propose solves Problem 1 regardless of the way the sensors are indexed, as long as the system and graph satisfy Condition 1. However, the question of appropriately ordering the sensors (or including redundancy) will become important when dealing with stochastic systems or with sensor failures.*

2.5.2 Observer Design

Using the matrix $\mathcal{T}$ identified in Proposition 2.5.1, we perform the coordinate transformation $\mathbf{x}[k] = \mathcal{T}\mathbf{z}[k]$ to obtain

$$\begin{aligned} \mathbf{z}[k+1] &= \bar{\mathbf{A}}\mathbf{z}[k], \\ \mathbf{y}_i[k] &= \bar{\mathbf{C}}_i\mathbf{z}[k], \quad \forall i \in \{1, \cdots, N\}, \end{aligned} \tag{2.6}$$

where $\bar{\mathbf{A}} = \mathcal{T}^{-1}\mathbf{A}\mathcal{T}$ and $\bar{\mathbf{C}}_i = \mathbf{C}_i\mathcal{T} = \begin{bmatrix} \mathbf{C}_{i1} & \cdots & \mathbf{C}_{ii} \mid \mathbf{0} \end{bmatrix}$ are given by (2.5). The vector $\mathbf{z}[k]$ assumes the following structure (commensurate with the structure of $\bar{\mathbf{A}}$ in (2.5)):

$$\mathbf{z}[k] = \begin{bmatrix} \mathbf{z}^{(1)}[k]^T & \cdots & \mathbf{z}^{(N)}[k]^T & \mathbf{z}_{\mathcal{U}}[k]^T \end{bmatrix}^T. \tag{2.7}$$

Here, $\mathbf{z}_{\mathcal{U}}[k]$ is precisely the unobservable portion of the state $\mathbf{z}[k]$, with respect to the pair $(\mathbf{A}, \mathbf{C})$. We call $\mathbf{z}^{(j)}[k] \in \mathbb{R}^{o_j}$ the j-th sub-state, and $\mathbf{z}_{\mathcal{U}}[k]$ the unobservable sub-state. Notice that based on the multi-sensor observable canonical decomposition, there is a one-to-one correspondence between a node j and its associated sub-state $\mathbf{z}^{(j)}[k]$. Accordingly, node j is viewed as the source of information of its corresponding sub-state $\mathbf{z}^{(j)}[k]$, and is tasked with the responsibility of estimating this sub-state. We call node j the *source node* for sub-state j. For each of the N sub-states, we thus have a unique source node (based on the initial labeling of the nodes). However, there is no unique source of information for the unobservable sub-state $\mathbf{z}_{\mathcal{U}}[k]$, as this portion of the state does not correspond to the observable subspace of any of the nodes in the network. Each node will thus maintain an estimate of $\mathbf{z}_{\mathcal{U}}[k]$, which it updates as a linear function of its *own* estimates of each of the N sub-states $\mathbf{z}^{(j)}[k], \forall j \in \{1, 2, \cdots, N\}$.

Remark 2.5.5 *It should be noted that a given sub-state $\mathbf{z}^{(j)}[k]$ in equation (2.7) might be of zero dimension (i.e., the sub-state can be empty). For instance, this can happen if its corresponding source of information, namely node j, has no measurements, i.e., if $\mathbf{C}_j = \mathbf{0}$.*

First, based on equations (2.5), (2.6) and (2.7), we observe that the dynamics of the i-th sub-state are governed by the equations

$$\begin{aligned} \mathbf{z}^{(i)}[k+1] &= \mathbf{A}_{ii}\mathbf{z}^{(i)}[k] + \sum_{j=1}^{i-1} \mathbf{A}_{ij}\mathbf{z}^{(j)}[k], \\ \mathbf{y}_i[k] &= \mathbf{C}_{ii}\mathbf{z}^{(i)}[k] + \sum_{j=1}^{i-1} \mathbf{C}_{ij}\mathbf{z}^{(j)}[k]. \end{aligned} \tag{2.8}$$

The reader is referred to the proof of Proposition 2.5.1 in Section 2.11.1 for a mathematical description of the matrices appearing in (2.8). Note that the unobservable sub-state $\mathbf{z}_{\mathcal{U}}[k]$ is governed by the dynamics

$$\mathbf{z}_{\mathcal{U}}[k+1] = \mathbf{A}_{\mathcal{U}}\mathbf{z}_{\mathcal{U}}[k] + \sum_{j=1}^{N} \mathbf{A}_j\mathbf{z}^{(j)}[k], \tag{2.9}$$

where the matrices $\mathbf{A}_j$ describe the coupling that exists between the unobservable sub-state $\mathbf{z}_{\mathcal{U}}[k]$ and each of the N sub-states $\mathbf{z}^{(j)}[k]$. Define $\hat{\mathbf{z}}_i^{(j)}[k]$ as the estimate of the j-th sub-state maintained by the i-th node. The estimation policy adopted by the i-th node is as follows - it uses a Luenberger-style update rule for updating its associated sub-state estimate $\hat{\mathbf{z}}_i^{(i)}[k]$, and a consensus based scheme for updating its estimates of all other sub-states $\hat{\mathbf{z}}_i^{(j)}[k]$, where $j \in \{1, \cdots, N\} \setminus \{i\}$. Based on the dynamics (2.8), the Luenberger observer at node i is constructed as

$$\begin{aligned} \hat{\mathbf{z}}_i^{(i)}[k+1] &= \mathbf{A}_{ii}\hat{\mathbf{z}}_i^{(i)}[k] + \sum_{j=1}^{i-1} \mathbf{A}_{ij}\hat{\mathbf{z}}_i^{(j)}[k] \\ &\quad + \mathbf{L}_i\left(\mathbf{y}_i[k] - \left(\mathbf{C}_{ii}\hat{\mathbf{z}}_i^{(i)}[k] + \sum_{j=1}^{i-1} \mathbf{C}_{ij}\hat{\mathbf{z}}_i^{(j)}[k]\right)\right), \end{aligned} \tag{2.10}$$

where $\mathbf{L}_i \in \mathbb{R}^{o_i \times r_i}$ is a gain matrix which needs to be designed. For estimation of the j-th sub-state, where $j \in \{1, \cdots, N\} \setminus \{i\}$, the i-th node again mimics the first equation in (2.8), but this time relies on consensus dynamics of the form

$$\hat{\mathbf{z}}_i^{(j)}[k+1] = \underbrace{\mathbf{A}_{jj} \sum_{l \in \mathcal{N}_i} w_{il}^{(j)} \hat{\mathbf{z}}_l^{(j)}[k]}_{\text{consensus term}} + \underbrace{\sum_{l=1}^{j-1} \mathbf{A}_{jl} \hat{\mathbf{z}}_i^{(l)}[k]}_{\text{coupling term}}, \tag{2.11}$$

where $w_{il}^{(j)}$ is the weight the i-th node associates with the l-th node, for the estimation of the j-th sub-state. The weights are non-negative and satisfy

$$\sum_{l \in \mathcal{N}_i} w_{il}^{(j)} = 1, \quad \forall j \in \{1, \cdots, N\} \setminus \{i\}. \tag{2.12}$$

In equation (2.11), the first term is a standard consensus term, while the second term has been introduced specifically to account for the coupling that exists between a given sub-state j and sub-states 1 to $j-1$ (as given by (2.8)). Let $\hat{\mathbf{z}}_{i\mathcal{U}}[k]$ denote the estimate of the unobservable sub-state $\mathbf{z}_{\mathcal{U}}[k]$ maintained by the i-th node. Mimicking equation (2.9), each node i uses the following rule to update $\hat{\mathbf{z}}_{i\mathcal{U}}[k]$:

$$\hat{\mathbf{z}}_{i\mathcal{U}}[k+1] = \mathbf{A}_{\mathcal{U}} \hat{\mathbf{z}}_{i\mathcal{U}}[k] + \sum_{j=1}^{N} \mathbf{A}_j \hat{\mathbf{z}}_i^{(j)}[k]. \tag{2.13}$$

In summary, equations (2.10), (2.11) and (2.13) together form the observer for the state $\mathbf{z}[k] = \mathcal{T}^{-1}\mathbf{x}[k]$ maintained by each node i.

2.5.3 Error Dynamics at each Node

Define $\mathbf{e}_i^{(j)}[k] \triangleq \hat{\mathbf{z}}_i^{(j)}[k] - \mathbf{z}^{(j)}[k]$ as the error in estimation of the j-th sub-state by the i-th node. Using equations (2.8) and (2.10), we obtain the error in the Luenberger observer dynamics at the i-th node as

$$\mathbf{e}_i^{(i)}[k+1] = (\mathbf{A}_{ii} - \mathbf{L}_i \mathbf{C}_{ii})\, \mathbf{e}_i^{(i)}[k] + \sum_{j=1}^{i-1} (\mathbf{A}_{ij} - \mathbf{L}_i \mathbf{C}_{ij})\, \mathbf{e}_i^{(j)}[k]. \tag{2.14}$$

Similarly, noting that $\mathbf{A}_{jj} = \mathbf{A}_{jj}\sum_{l\in\mathcal{N}_i} w_{il}^{(j)}$ (based on equation (2.12)), and using equations (2.8) and (2.11), we obtain the following consensus error dynamics at node i, $\forall j \in \{1,\cdots,N\}\setminus\{i\}$:

$$\mathbf{e}_i^{(j)}[k+1] = \mathbf{A}_{jj}\sum_{l\in\mathcal{N}_i} w_{il}^{(j)}\mathbf{e}_l^{(j)}[k] + \sum_{l=1}^{j-1}\mathbf{A}_{jl}\mathbf{e}_i^{(l)}[k]. \tag{2.15}$$

Define $\mathbf{e}_{i\mathcal{U}}[k] \triangleq \hat{\mathbf{z}}_{i\mathcal{U}}[k] - \mathbf{z}_{\mathcal{U}}[k]$ as the error in estimation of the unobservable sub-state $\mathbf{z}_{\mathcal{U}}[k]$ by the i-th node. Using (2.9) and (2.13), we obtain the following error dynamics for the unobservable sub-state at node i:

$$\mathbf{e}_{i\mathcal{U}}[k+1] = \mathbf{A}_{\mathcal{U}}\mathbf{e}_{i\mathcal{U}}[k] + \sum_{j=1}^{N}\mathbf{A}_j\mathbf{e}_i^{(j)}[k]. \tag{2.16}$$

2.5.4 Analysis of the Estimation Scheme for Systems and Graphs Satisfying Condition 1

In this section, we present our main result, formally stated as follows.

Theorem 2.5.6 *Consider a system* $(\mathbf{A},\mathbf{C})$ *and graph* $\mathcal{G}$ *satisfying Condition 1. Let Assumption 2.5.1 hold true. Then, for each node* $i \in \{1,2,\cdots,N\}$*, there exists a choice of observer gain matrix* $\mathbf{L}_i$*, and consensus weights* $w_{il}^{(j)}$*,* $j \in \{1,2,\cdots,N\}\setminus\{i\}$*,* $l \in \mathcal{N}_i$*, such that the update rules given by equations (2.10), (2.11), and (2.13) form a distributed observer.*

Proof Consider the composite error in estimation of sub-state j by all of the nodes in $\mathcal{V}$, defined as

$$\mathbf{E}^{(j)}[k] \triangleq \begin{bmatrix} \mathbf{e}_1^{(j)}[k] \\ \mathbf{e}_2^{(j)}[k] \\ \vdots \\ \mathbf{e}_N^{(j)}[k] \end{bmatrix}. \tag{2.17}$$

We will prove that $\mathbf{E}^{(j)}[k]$ converges to zero asymptotically $\forall j \in \{1,\cdots,N\}$ (recall that there are precisely N nodes in the network, each responsible for estimating a certain sub-state). We prove by induction on j. Consider the base case $j = 1$, i.e., the estimation of the first sub-state. Let the index set $\{1, k_1, k_2, \cdots, k_{N-1}\}$ represent a topological ordering[8] consistent with a spanning tree rooted at node 1 (the source node for sub-state 1). Note that based on Assumption 2.5.1, it is always possible to find such a spanning tree. Next, consider the composite error vector

$$\bar{\mathbf{E}}^{(1)}[k] = \begin{bmatrix} \mathbf{e}_1^{(1)}[k] \\ \mathbf{e}_{k_1}^{(1)}[k] \\ \vdots \\ \mathbf{e}_{k_{N-1}}^{(1)}[k] \end{bmatrix} = \begin{bmatrix} \mathbf{e}_1^{(1)}[k] \\ \tilde{\mathbf{E}}^{(1)}[k] \end{bmatrix}, \tag{2.18}$$

where $\tilde{\mathbf{E}}^{(1)}[k] \triangleq \left[\mathbf{e}_{k_1}^{(1)}[k]^T \cdots \mathbf{e}_{k_{N-1}}^{(1)}[k]^T \right]^T$. Note that $\bar{\mathbf{E}}^{(1)}[k]$ is simply a permutation of the rows of $\mathbf{E}^{(1)}[k]$. Based on the error dynamics equations given by (2.14) and (2.15), we obtain

$$\underbrace{\begin{bmatrix} \mathbf{e}_1^{(1)}[k+1] \\ \tilde{\mathbf{E}}^{(1)}[k+1] \end{bmatrix}}_{\bar{\mathbf{E}}^{(1)}[k+1]} = \underbrace{\begin{bmatrix} (\mathbf{A}_{11} - \mathbf{L}_1\mathbf{C}_{11}) & \mathbf{0} \\ \mathbf{W}_{21}^1 \otimes \mathbf{A}_{11} & \mathbf{W}_{22}^1 \otimes \mathbf{A}_{11} \end{bmatrix}}_{\mathbf{M}_1} \underbrace{\begin{bmatrix} \mathbf{e}_1^{(1)}[k] \\ \tilde{\mathbf{E}}^{(1)}[k] \end{bmatrix}}_{\bar{\mathbf{E}}^{(1)}[k]}, \tag{2.19}$$

where the entries of the weight matrix $\mathbf{W}^1 = \begin{bmatrix} \mathbf{W}_{21}^1 & \mathbf{W}_{22}^1 \end{bmatrix}$ are populated by the appropriate weights defined by equation (2.15) (note that $\mathbf{W}^1 \in \mathbb{R}^{(N-1)\times N}$ and $\mathbf{W}_{21}^1$ is the first column of $\mathbf{W}^1$). Notice that $sp(\mathbf{M}_1) = sp(\mathbf{A}_{11} - \mathbf{L}_1\mathbf{C}_{11}) \cup sp(\mathbf{W}_{22}^1 \otimes \mathbf{A}_{11})$. By construction, the pair $(\mathbf{A}_{11}, \mathbf{C}_{11})$ is observable. Thus, it is always possible to find a gain matrix $\mathbf{L}_1$ such that $(\mathbf{A}_{11} - \mathbf{L}_1\mathbf{C}_{11})$ is Schur stable. Next, we impose the constraint that for the estimation of sub-state 1, non-zero consensus weights are assigned to only the branches of the spanning tree consistent with the ordering $\{1, k_1, k_2, \cdots, k_{N-1}\}$, i.e., a node listens to only its parent in such a tree. In this way, $\mathbf{W}_{22}^1$ becomes lower triangular with eigenvalues equal to zero, without violating the

[8] Such an ordering results when a standard Breadth-First Search (BFS) [60] algorithm is applied to the graph $\mathcal{G}$, with node 1 as the root node of the tree. Specifically, the order represents the order in which the nodes are added to the spanning tree when the BFS algorithm is implemented, i.e., node k_1 would be added first, followed by node k_2 and so on. This ordering naturally leads to a lower triangular adjacency matrix for the constructed spanning tree.

stochasticity condition imposed on $\mathbf{W}^1$ by equation (2.12). We conclude that by an appropriate choice of consensus weights, we can achieve $\Lambda_U(\mathbf{W}^1_{22} \otimes \mathbf{A}_{11}) = \emptyset$ (even if $\Lambda_U(\mathbf{A}_{11}) \neq \emptyset$). Here, we use the result that if $\mathbf{A} \in \mathbb{R}^{n\times n}$ and $\mathbf{B} \in \mathbb{R}^{m\times m}$, then the eigenvalues of the Kronecker product $\mathbf{A} \otimes \mathbf{B} \in \mathbb{R}^{mn\times mn}$ are the mn numbers $\lambda_i(\mathbf{A})\lambda_j(\mathbf{B}), (i = 1, \cdots, n; j = 1, \cdots, m)$ [61].

It follows that $\mathbf{M}_1$ can be made Schur stable, and hence $\lim_{k\to\infty} \bar{\mathbf{E}}^{(1)}[k] = \mathbf{0}$, implying $\lim_{k\to\infty} \mathbf{E}^{(1)}[k] = \mathbf{0}$ (one is just a permutation of the other). Thus, the base case is proven. Next, suppose that $\mathbf{E}^{(j)}[k]$ converges to zero asymptotically $\forall j \in \{1, \cdots, p-1\}$, where $1 \leq p-1 \leq N-1$. Consider the following composite error vector for the p-th sub-state:

$$\bar{\mathbf{E}}^{(p)}[k] = \begin{bmatrix} \mathbf{e}^{(p)}_p[k] \\ \mathbf{e}^{(p)}_{m_1}[k] \\ \vdots \\ \mathbf{e}^{(p)}_{m_{N-1}}[k] \end{bmatrix} = \begin{bmatrix} \mathbf{e}^{(p)}_p[k] \\ \tilde{\mathbf{E}}^{(p)}[k] \end{bmatrix}, \tag{2.20}$$

where the index set $\{p, m_1, m_2, \cdots, m_{N-1}\}$ represents a topological ordering of the nodes of $\mathcal{V}$ to obtain a spanning tree rooted at node p (the source node for sub-state p), and $\tilde{\mathbf{E}}^{(p)}[k] \triangleq \left[\mathbf{e}^{(p)}_{m_1}[k]^T \cdots \mathbf{e}^{(p)}_{m_{N-1}}[k]^T\right]^T$. From the error dynamics equations given by (2.14) and (2.15), we obtain

$$\bar{\mathbf{E}}^{(p)}[k+1] = \mathbf{M}_p\bar{\mathbf{E}}^{(p)}[k] + \sum_{l=1}^{p-1} \mathbf{H}_{pl}\bar{\mathbf{E}}^{(pl)}[k], \tag{2.21}$$

where

$$\mathbf{M}_p = \begin{bmatrix} (\mathbf{A}_{pp} - \mathbf{L}_p\mathbf{C}_{pp}) & \mathbf{0} \\ \mathbf{W}^p_{21} \otimes \mathbf{A}_{pp} & \mathbf{W}^p_{22} \otimes \mathbf{A}_{pp} \end{bmatrix}, \tag{2.22}$$

$$\mathbf{H}_{pl} = diag\left(\mathbf{A}_{pl} - \mathbf{L}_p\mathbf{C}_{pl}, \mathbf{I}_{N-1} \otimes \mathbf{A}_{pl}\right), \tag{2.23}$$

$$\bar{\mathbf{E}}^{(pl)}[k] = \begin{bmatrix} \mathbf{e}^{(l)}_p[k] \\ \mathbf{e}^{(l)}_{m_1}[k] \\ \vdots \\ \mathbf{e}^{(l)}_{m_{N-1}}[k] \end{bmatrix}. \tag{2.24}$$

By following the same reasoning as the base case, one concludes that $\mathbf{M}_p$ can be made Schur stable by appropriate choices of the observer gain matrix $\mathbf{L}_p$, and consensus weight matrix $\mathbf{W}^p = \begin{bmatrix} \mathbf{W}^p_{21} & \mathbf{W}^p_{22} \end{bmatrix}$ (note that $\mathbf{W}^p \in \mathbb{R}^{(N-1)\times N}$ and $\mathbf{W}^p_{21}$ is the first column of $\mathbf{W}^p$). Specifically, non-zero weights are assigned in $\mathbf{W}^p$ only on the branches of the tree rooted at node p, consistent with the topological ordering. Notice that $\bar{\mathbf{E}}^{(pl)}[k]$ is simply a permutation of the rows of $\mathbf{E}^{(l)}[k]$ (permuted to match the order of indices in $\bar{\mathbf{E}}^{(p)}[k]$). Further, based on our induction hypothesis, $\mathbf{E}^{(l)}[k]$ converges to zero asymptotically (since $1 \leq l \leq p-1$). Thus, by Input to State Stability (ISS), we conclude that $\bar{\mathbf{E}}^{(p)}[k]$, and hence $\mathbf{E}^{(p)}[k]$, converges to zero asymptotically. We have thus proven that the composite estimation error for every sub-state asymptotically approaches zero, i.e., $\lim_{k\to\infty} \mathbf{e}^{(j)}_i[k] = \mathbf{0}, \forall i, j \in \{1, \cdots N\}$.

Finally, consider the error in estimation of the unobservable sub-state $\mathbf{z}_{\mathcal{U}}[k]$ (given by equation (2.16)). As the system and graph under consideration satisfy Condition 1 and Assumption 2.5.1, it must be that the pair $(\mathbf{A}, \mathbf{C})$ is detectable. Thus, based on Proposition 2.5.1, the matrix $\mathbf{A}_{\mathcal{U}}$ in (2.16) must be stable. Invoking ISS, we have that $\lim_{k\to\infty} \mathbf{e}_{i\mathcal{U}}[k] = \mathbf{0}, \forall i \in \{1, \cdots, N\}$. Thus, every node in the network can asymptotically estimate $\mathbf{z}[k]$, and hence $\mathbf{x}[k]$, as $\mathbf{x}[k] = \mathcal{T}\mathbf{z}[k]$. ■

2.5.5 A Compact Representation of the Proposed Observer

In this section, we combine the update equations (2.10), (2.11) and (2.13) to obtain a compact representation of our distributed observer. To do so, we need to first introduce some notation. Accordingly, let $\mathbf{B}_j = \begin{bmatrix} \mathbf{0} \cdots \mathbf{I}_{o_j} \cdots \mathbf{0} \end{bmatrix}$ be the matrix that extracts the j-th sub-state from the transformed state vector $\mathbf{z}[k]$, i.e., $\mathbf{z}^{(j)}[k] = \mathbf{B}_j\mathbf{z}[k]$. Similarly, let $\mathbf{B}_{\mathcal{U}}$ be such that $\mathbf{z}_{\mathcal{U}}[k] = \mathbf{B}_{\mathcal{U}}\mathbf{z}[k]$. Define $\mathbb{B} \triangleq diag(\mathbf{B}_1, \cdots, \mathbf{B}_N, \mathbf{B}_{\mathcal{U}})$. Next, notice that the transformed system matrix $\bar{\mathbf{A}}$ in equation (2.5) can be written as $\bar{\mathbf{A}} = \bar{\mathcal{A}}_1 + \bar{\mathcal{A}}_2$, where $\bar{\mathcal{A}}_2 = diag(\mathbf{A}_{11}, \cdots, \mathbf{A}_{NN}, \mathbf{A}_{\mathcal{U}})$, and $\bar{\mathcal{A}}_1$ is a block lower-triangular matrix given by $\bar{\mathbf{A}} - \bar{\mathcal{A}}_2$. Let $\mathbf{w}_{il}$ (where $l \in \mathcal{N}_i \setminus \{i\}$) be the vector of weights node i associates with a neighbor l for the estimation of the transformed state $\mathbf{z}[k]$. Based on our estimation scheme, note that at any given time-step k, node i does not use the estimates received from its neighbors at time-step k for estimating $\mathbf{z}^{(i)}[k]$ and $\mathbf{z}_{\mathcal{U}}[k]$, and hence these weight vectors assume the following form: $\mathbf{w}_{il} = \left[w_{il}^{(1)}, \cdots, w_{il}^{(i-1)}, 0, w_{il}^{(i+1)}, \cdots, w_{il}^{(N)}, 0\right]^T, \forall l \in \mathcal{N}_i \setminus \{i\}$. Also, notice that the element $w_{il}^{(j)}$ is not present in the vector if the j-th sub-state is empty (i.e., of dimension 0). Similarly, let $\mathbf{w}_{ii}$ be a vector with a '1' in the elements corresponding to the i-th sub-state and the unobservable sub-state $\mathbf{z}_{\mathcal{U}}[k]$, and zeroes at all other positions. Finally, defining $\mathbb{H}_i \triangleq \left[\mathbf{0}^T \cdots \mathbf{L}_i{}^T \cdots \mathbf{0}^T\right]^T$, using equations (2.10), (2.11), and (2.13), and noting that $\mathbf{z}[k] = \mathcal{T}^{-1}\mathbf{x}[k]$, we obtain the following overall state estimate update rule at node i:

$$\boxed{\hat{\mathbf{x}}_i[k+1] = \mathcal{T}\bar{\mathcal{A}}_1\mathcal{T}^{-1}\hat{\mathbf{x}}_i[k] + \underbrace{\mathcal{T}\mathbb{H}_i(\mathbf{y}_i[k] - \mathbf{C}_i\hat{\mathbf{x}}_i[k])}_{\text{innovation term}} + \underbrace{\sum_{l\in\mathcal{N}_i}\mathbb{G}_{il}\hat{\mathbf{x}}_l[k]}_{\text{“consensus term”}},} \tag{2.25}$$

where $\hat{\mathbf{x}}_i[k]$ denotes the estimate of the state $\mathbf{x}[k]$ maintained by node i, and $\mathbb{G}_{il} = \mathcal{T}\bar{\mathcal{A}}_2\mathbb{B}\left(\mathbf{w}_{il} \otimes \mathcal{T}^{-1}\right)$.

Remark 2.5.7 *From the structure of our overall estimator at node i, as represented by equation (2.25), it is easy to see that the estimator maintained at each node has dimension equal to n (i.e., equal to that of the state). Thus, our approach alleviates the need to construct augmented observers such as those considered in [1, 30].*

Remark 2.5.8 *Note that all the transformation and gain matrices appearing in (2.25) can be computed offline during a centralized design phase. Thus, although the observer design and the subsequent analysis were done in the $\mathbf{z}[k]$ coordinate system, no inversion from $\mathbf{z}[k]$ to $\mathbf{x}[k]$ is necessary while implementing (2.25) during run-time, i.e., the nodes directly exchange their estimates of the actual state $\mathbf{x}[k]$, and not $\mathbf{z}[k]$.*

2.5.6 Summary of the Estimation Scheme for Systems and Graphs Satisfying Condition 1

The proposed distributed observer scheme for systems and graphs satisfying Condition 1 (under the assumption that the graph $\mathcal{G}$ is strongly connected) can be broadly decomposed into two main phases, namely the design phase and the distributed estimation phase. For clarity, we briefly enumerate the steps associated with each of these phases.

Design Phase:

- Each node of the graph is assigned a unique integer between 1 to N. Based on this numbering, the multi-sensor observable canonical decomposition (as outlined in the proof of Proposition 2.5.1) is performed, yielding the state $\mathbf{z}[k] = \mathcal{T}^{-1}\mathbf{x}[k]$.

- Based on this transformation, each node is associated with a sub-state of $\mathbf{z}[k]$ that it is responsible for estimating. Recall that there are precisely N sub-states, one corresponding to each node in the network; some of these sub-states might be empty.

- For the estimation of a given sub-state, we construct a spanning tree rooted at the specific node which acts as the source of information for that sub-state. The resulting spanning tree guides the construction of the consensus weight matrix to be used for the estimation of that particular sub-state. We construct one spanning tree for the estimation of each non-empty sub-state.

- Based on the constructed consensus weight matrices, and the Luenberger observer gains $\mathbf{L}_i$, the matrices $\mathcal{T}\bar{\mathcal{A}}_1\mathcal{T}^{-1}$, $\mathcal{T}\mathbb{H}_i$ and $\mathbb{G}_{il}$ in (2.25) are computed for each node $i \in \mathcal{V}$.

Estimation Phase (Run-time):

- Each node employs a Luenberger observer for constructing an estimate of its corresponding sub-state, and runs consensus dynamics for estimating the sub-states corresponding to the remaining nodes in the network. Summarily, a node implements (2.25) for estimating $\mathbf{x}[k]$.

Remark 2.5.9 *While the observer design procedure we have outlined (involving the multi-sensor decomposition, design of local observer gains, construction of spanning trees and selection of consensus weights) can be readily implemented in a centralized manner, it may also be possible to perform these steps in a distributed fashion. This would require the nodes to assign themselves unique identifiers (or labels) and execute the multi-sensor decomposition in a round-robin fashion, followed by a distributed construction of spanning trees. However, at present, the multi-sensor decomposition appears to be the most expensive portion of such an implementation (in terms of co-ordination and communication). In Section 2.6, we show that for systems and graphs that possess the additional structure described by Condition 2, we can avoid such a decomposition and obtain a scheme that permits an efficient distributed implementation (in both the design and run-time phases) at the potential cost of increasing the dimension of the observer.*

Having established our approach for all systems and strongly connected graphs satisfying Condition 1, we now briefly describe the extension of our strategy to arbitrary directed networks.

2.5.7 Extension to General Directed Networks

Our distributed observer design can be extended to general networks (satisfying Condition 1 but not necessarily Assumption 2.5.1) by first decomposing $\mathcal{G}$ into its strong components, and identifying each of the source components. Next, within a given source component, one simply follows the observer design procedure outlined in Section 2.5.2 for a strongly connected graph, to obtain an estimator of the form (2.25) for each node within the source component. Define $\mathcal{S} \triangleq \bigcup_{i=1}^{p} \mathcal{V}_i$ to be the set of all nodes that belong to the source components of $\mathcal{G}$. Let each node in $i \in \mathcal{V} \setminus \mathcal{S}$ employ a pure consensus strategy of the form

$$\hat{\mathbf{x}}_i[k+1] = \mathbf{A} \sum_{j \in \mathcal{N}_i} w_{ij} \hat{\mathbf{x}}_j[k], \tag{2.26}$$

where $\hat{\mathbf{x}}_i[k]$ represents an estimate of the state maintained by the i-th node. The weights w_{ij} are non-negative and satisfy

$$\sum_{j \in \mathcal{N}_i} w_{ij} = 1, \quad \forall i \in \mathcal{V} \setminus \mathcal{S}. \tag{2.27}$$

The design of consensus weights for the nodes in $\mathcal{V} \setminus \mathcal{S}$ is based on the observation that the set $\mathcal{V} \setminus \mathcal{S}$ can be spanned by a disjoint union of trees rooted in $\mathcal{S}$. By assigning consensus weights to only the branches of these trees (without violating the stochasticity condition imposed by equation (2.27)), one obtains stable estimation error dynamics for each of the nodes in $\mathcal{V} \setminus \mathcal{S}$ (the details are similar to the proof of Theorem 2.5.6). The above strategy readily leads to the following result.

Theorem 2.5.10 *Consider a system* $(\mathbf{A}, \mathbf{C})$ *and graph* $\mathcal{G}$ *satisfying Condition 1. Let each node in* $\mathcal{S}$ *run an observer of the form (2.25), and each node in* $\mathcal{V} \setminus \mathcal{S}$ *run the consensus dynamics given by (2.26). Then, there exists a choice of consensus weights and observer gain matrices that results in a distributed observer.*

As discussed in Remark 2.5.9, our distributed observer design starts with the multi-sensor observable decomposition described in Proposition 2.5.1, which transforms the system into a form that identifies the sub-states that each node is responsible for estimating. This decomposition requires knowledge of the measurement matrices of each node, and thus is most amenable to a centralized implementation (a centralized design phase is commonly assumed in the existing literature on distributed observers, e.g., [1,27–32]). In the next section, we show that for systems and networks satisfying Condition 2, the design of the observer itself can be readily done in a distributed manner. However, before we conclude this section, it is instructive to note the following.

Remark 2.5.11 ***(The effect of noise)****: For scenarios where the system and measurements are affected by noise, a desirable objective is to formulate a distributed estimation strategy that guarantees bounded mean square estimation error. As we mention in Section 2.2.1, most of the literature that attempts to address this question does so by either resorting to two-time-scale algorithms or LMI-based frameworks. The only papers we are aware of that guarantee bounded mean square error under the most general conditions on the system and network are [1] and [30] (like our present approach, the analysis in these papers is also conducted for a noiseless model). While the method we developed in Section 2.5 provides the same guarantees of stability against i.i.d. noise with bounded second moments as those works,*[9] *our approach offers the additional advantage of requiring no state augmentation. However, the scheme that we have proposed will not, in general, be optimal in terms of minimizing the mean square estimation error, due to the fact that each sub-state is directly estimated by a single node in our multi-sensor observable decomposition, and due to the tree structure that we construct for the rest of the nodes in the network. Extending our approach to incorporate redundancy in order to further minimize the mean square estimation error is an important avenue for future work. In the next section, we describe a slightly modified scheme for systems that satisfy Condition* 2 *given earlier in the chapter, which allows us to assign multiple nodes to simultaneously be responsible for estimating the same states, and facilitates the incorporation of redundancy that can, among other things, allow resilience to node failures and attacks.*

2.6 Estimation Scheme for systems and graphs satisfying Condition 2

Recall that for systems and graphs satisfying Condition 2, for each eigenvalue of the plant, there is at least one node in each source component that can detect that

[9] Such guarantees are also provided by the method we develop in Section 2.6 for systems and graphs satisfying Condition 2.

eigenvalue. As we will show, this fact allows each node in the network to identify the sub-states it is responsible for estimating, without having to exchange any information with neighbors.

To this end, let $\mathcal{T}$ be a non-singular transformation matrix which transforms $\mathbf{A}$ into its Jordan canonical form $\mathbf{J}$, i.e., $\mathbf{A} = \mathcal{T}\mathbf{J}\mathcal{T}^{-1}$. With $\mathbf{z}[k] = \mathcal{T}^{-1}\mathbf{x}[k]$, the dynamics (2.1) are transformed into the form

$$\begin{aligned} \mathbf{z}[k+1] &= \mathbf{J}\mathbf{z}[k], \\ \mathbf{y}_i[k] &= \bar{\mathbf{C}}_i\mathbf{z}[k], \quad \forall i \in \{1, \cdots, N\} \end{aligned} \tag{2.28}$$

where $\mathbf{J} = \mathcal{T}^{-1}\mathbf{A}\mathcal{T}$ and $\bar{\mathbf{C}}_i = \mathbf{C}_i\mathcal{T}$.[10] Notice that this transformation relies only on the knowledge of the system matrix $\mathbf{A}$ (which is assumed to be known by all of the nodes). Hence, all nodes can perform this transformation *in parallel* (e.g., by using an agreed-upon convention for ordering the eigenvalues and corresponding eigenvectors). We denote the eigenvalues of $\mathbf{J}$ (which are the same as those of $\mathbf{A}$) by $\lambda_1, \cdots, \lambda_\gamma$, where γ represents the number of distinct eigenvalues of $\mathbf{A}$. Let $\mathbf{J} = diag(\mathbf{J}_1, \cdots, \mathbf{J}_\gamma)$, where we group all of the Jordan blocks associated with $\lambda_j \in sp(\mathbf{J})$ into the block diagonal matrix $\mathbf{J}_j \in \mathbb{R}^{a_{\mathbf{J}}(\lambda_j) \times a_{\mathbf{J}}(\lambda_j)}$. The portion of the state $\mathbf{z}[k]$ associated with the eigenvalue λ_j is termed as the sub-state $\mathbf{z}^{(j)}[k] \in \mathbb{R}^{a_{\mathbf{J}}(\lambda_j)}$. Let $\hat{\mathbf{z}}_i^{(j)}[k]$ represent the estimate of $\mathbf{z}^{(j)}[k]$ maintained by node i. Note that if each node in the network can accurately estimate $\mathbf{z}[k]$, then they can also estimate $\mathbf{x}[k]$ using the relation $\mathbf{x}[k] = \mathcal{T}\mathbf{z}[k]$. In view of this, we now develop a scheme for estimating $\mathbf{z}[k]$.

2.6.1 Distributed Observer Design

Design of Local Luenberger Observers

Let $\mathcal{O}_i$ represent the set of detectable eigenvalues of node i. For estimating the sub-states corresponding to the eigenvalues in $\mathcal{O}_i$, node i constructs a simple Luenberger

[10] Note that the matrices $\mathcal{T}$ and $\bar{\mathbf{C}}_i$ in (2.28) are in general different from those in (2.5); we adopt this abuse of notation to avoid cluttering the exposition with additional symbols.

observer using its own measurements. To this end, permute the states $\mathbf{z}[k]$ in (2.28) to obtain

$$\begin{aligned} \underbrace{\begin{bmatrix} \mathbf{z}_{\mathcal{O}_i}[k+1] \\ \mathbf{z}_{\mathcal{UO}_i}[k+1] \end{bmatrix}}_{\bar{\mathbf{z}}_i[k+1]} &= \underbrace{\begin{bmatrix} \bar{\mathbf{J}}_{\mathcal{O}_i} & \mathbf{0} \\ \mathbf{0} & \bar{\mathbf{J}}_{\mathcal{UO}_i} \end{bmatrix}}_{\bar{\mathcal{J}}_i} \underbrace{\begin{bmatrix} \mathbf{z}_{\mathcal{O}_i}[k] \\ \mathbf{z}_{\mathcal{UO}_i}[k] \end{bmatrix}}_{\bar{\mathbf{z}}_i[k]}, \\ \mathbf{y}_i[k] &= \underbrace{\begin{bmatrix} \bar{\mathbf{C}}_{\mathcal{O}_i} & \bar{\mathbf{C}}_{\mathcal{UO}_i} \end{bmatrix}}_{\bar{\mathcal{C}}_i} \bar{\mathbf{z}}_i[k]. \end{aligned} \tag{2.29}$$

The permuted state $\bar{\mathbf{z}}_i[k]$ will be represented by $\mathbf{z}[k] = \mathbb{P}_i\bar{\mathbf{z}}_i[k]$, where $\mathbb{P}_i$ is an appropriate permutation matrix. In the above equations, $\bar{\mathbf{J}}_{\mathcal{O}_i}$ consists of all Jordan blocks corresponding to the detectable eigenvalues of node i, and $\bar{\mathbf{J}}_{\mathcal{UO}_i}$ denotes the collection of Jordan blocks corresponding to the undetectable eigenvalues of node i. Similarly, $\bar{\mathbf{C}}_{\mathcal{O}_i}$ contains the columns of $\bar{\mathbf{C}}_i$ corresponding to the matrix $\bar{\mathbf{J}}_{\mathcal{O}_i}$, with an analogous definition for $\bar{\mathbf{C}}_{\mathcal{UO}_i}$. The sub-states corresponding to the detectable and undetectable eigenvalues of node i are grouped into the composite vectors $\mathbf{z}_{\mathcal{O}_i}[k] \in \mathbb{R}^{o_i}$ and $\mathbf{z}_{\mathcal{UO}_i}[k]$ respectively.

Based on (2.29), notice that the output $\mathbf{y}_i[k]$ is affected by elements of $\mathbf{z}_{\mathcal{UO}_i}[k]$ (through $\bar{\mathbf{C}}_{\mathcal{UO}_i}$) and thus we will estimate those elements as well in order to recover $\mathbf{z}_{\mathcal{O}_i}[k]$. To this end, let $\bar{\mathbf{T}}_i$ be a non-singular matrix which performs an observable canonical decomposition of the pair $(\bar{\mathbf{J}}_{\mathcal{UO}_i}, \bar{\mathbf{C}}_{\mathcal{UO}_i})$ in (2.29). Consider the following transformation matrix:

$$\mathbf{T}_i = \begin{bmatrix} \mathbf{I}_{o_i} & \mathbf{0} \\ \mathbf{0} & \bar{\mathbf{T}}_i \end{bmatrix}. \tag{2.30}$$

Define the coordinate transformation $\bar{\mathbf{z}}_i[k] = \mathbf{T}_i\mathbf{v}_i[k]$ (this transformation is specific to node i). Based on this transformation, and equations (2.29) and (2.30), the dynamics at node i can be reformulated as

$$\begin{aligned}
\underbrace{\begin{bmatrix} \mathbf{z}_{\mathcal{O}_i}[k+1] \\ \mathbf{w}_{\mathcal{O}_i}[k+1] \\ \mathbf{w}_{\mathcal{UO}_i}[k+1] \end{bmatrix}}_{\mathbf{v}_i[k+1]} &= \underbrace{\left[\begin{array}{c|cc} \bar{\mathbf{J}}_{\mathcal{O}_i} & \multicolumn{2}{c}{\mathbf{0}} \\ \hline \mathbf{0} & \mathbf{G}_{\mathcal{O}_i} & \mathbf{0} \\ & \star & \mathbf{G}_{\mathcal{UO}_i} \end{array}\right]}_{\mathbf{T}_i^{-1}\bar{\mathcal{J}}_i\mathbf{T}_i} \underbrace{\begin{bmatrix} \mathbf{z}_{\mathcal{O}_i}[k] \\ \mathbf{w}_{\mathcal{O}_i}[k] \\ \mathbf{w}_{\mathcal{UO}_i}[k] \end{bmatrix}}_{\mathbf{v}_i[k]}, \\
\mathbf{y}_i[k] &= \underbrace{\left[\begin{array}{c|cc} \bar{\mathbf{C}}_{\mathcal{O}_i} & \mathbf{H}_{\mathcal{O}_i} & \mathbf{0} \end{array}\right]}_{\bar{\mathcal{C}}_i\mathbf{T}_i} \mathbf{v}_i[k],
\end{aligned} \tag{2.31}$$

where

$$\begin{aligned}
\bar{\mathbf{T}}_i^{-1}\bar{\mathbf{J}}_{\mathcal{UO}_i}\bar{\mathbf{T}}_i &= \begin{bmatrix} \mathbf{G}_{\mathcal{O}_i} & \mathbf{0} \\ \star & \mathbf{G}_{\mathcal{UO}_i} \end{bmatrix}, \\
\bar{\mathbf{C}}_{\mathcal{UO}_i}\bar{\mathbf{T}}_i &= \begin{bmatrix} \mathbf{H}_{\mathcal{O}_i} & \mathbf{0} \end{bmatrix}.
\end{aligned} \tag{2.32}$$

Define

$$\mathbb{J}_i \triangleq diag(\bar{\mathbf{J}}_{\mathcal{O}_i}, \mathbf{G}_{\mathcal{O}_i}), \;\; \mathbb{F}_i \triangleq \begin{bmatrix} \bar{\mathbf{C}}_{\mathcal{O}_i} & \mathbf{H}_{\mathcal{O}_i} \end{bmatrix}, \tag{2.33}$$

and $\mathbf{s}_i[k] \triangleq \begin{bmatrix} \mathbf{z}_{\mathcal{O}_i}{}^T[k] & \mathbf{w}_{\mathcal{O}_i}{}^T[k] \end{bmatrix}^T$. Based on the dynamics (2.31), the local Luenberger observer maintained by node i for estimating $\mathbf{z}_{\mathcal{O}_i}[k]$ has the form

$$\hat{\mathbf{s}}_i[k+1] = \mathbb{J}_i\hat{\mathbf{s}}_i[k] + \mathbb{L}_i(\mathbf{y}_i[k] - \mathbb{F}_i\hat{\mathbf{s}}_i[k]), \tag{2.34}$$

where $\mathbb{L}_i$ is a gain matrix which needs to be designed for node i and $\hat{\mathbf{s}}_i[k]$ is the estimate of $\mathbf{s}_i[k]$ maintained by node i. Using (2.34), $\hat{\mathbf{z}}_{\mathcal{O}_i}[k]$ can then be updated as $\hat{\mathbf{z}}_{\mathcal{O}_i}[k+1] = \begin{bmatrix} \mathbf{I}_{o_i} & \mathbf{0} \end{bmatrix} \hat{\mathbf{s}}_i[k+1]$.

Based on the (local) transformation (2.31) and the (local) observer (2.34), we obtain the following result.

Lemma 2.6.1 *For a system* $(\mathbf{A}, \mathbf{C})$ *and graph* $\mathcal{G}$ *satisfying Condition 2, let every node* $i \in \mathcal{V}$ *run a Luenberger observer of the form (2.34). Then, there exists a choice of observer gain* $\mathbb{L}_i$*, which can be designed locally, such that for each* $\lambda_j \in \mathcal{O}_i$*,* $\lim_{k\to\infty} \|\hat{\mathbf{z}}_i^{(j)}[k] - \mathbf{z}^{(j)}[k]\| = 0$.

Proof The proof follows straightforwardly by noting that $(\mathbb{J}_i, \mathbb{F}_i)$ defined in (2.33) is detectable (since $\bar{\mathbf{J}}_{\mathcal{O}_i}$ and $\mathbf{G}_{\mathcal{O}_i}$ do not share any eigenvalues, and each of the pairs $(\bar{\mathbf{J}}_{\mathcal{O}_i}, \bar{\mathbf{C}}_{\mathcal{O}_i})$ and $(\mathbf{G}_{\mathcal{O}_i}, \mathbf{H}_{\mathcal{O}_i})$ are detectable, by construction). ■

Having established that each node $i \in \mathcal{V}$ can asymptotically recover $\mathbf{z}_{\mathcal{O}_i}[k]$ in (2.29) purely locally, we now devise a method that allows each node to estimate the sub-states corresponding to the locally undetectable eigenvalues.

Consensus dynamics

Consider an eigenvalue $\lambda_j \in \mathcal{UO}_i$ (recall $\mathcal{UO}_i$ represents the set of undetectable eigenvalues of node i). For such an eigenvalue, node i has to rely on the information received from its neighbors in order to estimate $\mathbf{z}^{(j)}[k]$. To this end, we propose the following consensus strategy to be followed by every node $i \in \mathcal{V} \setminus \mathcal{S}_j$ for updating their respective estimates of $\mathbf{z}^{(j)}[k]$:

$$\hat{\mathbf{z}}_i^{(j)}[k+1] = \mathbf{J}_j \sum_{l \in \mathcal{N}_i} w_{il}^{(j)} \hat{\mathbf{z}}_l^{(j)}[k], \tag{2.35}$$

where $w_{il}^{(j)}$ is the weight the i-th node associates with the l-th node for the estimation of the j-th sub-state (recall that $\mathcal{S}_j$ denotes the set of root nodes that can detect λ_j). Each weight is non-negative and satisfies

$$\sum_{l \in \mathcal{N}_i} w_{il}^{(j)} = 1, \quad \forall \lambda_j \in \mathcal{UO}_i. \tag{2.36}$$

Let $\mathcal{UO}_i = \{\lambda_{n_1}, \cdots, \lambda_{n_{\gamma_i}}\}$, where $\gamma_i = |\mathcal{UO}_i| = \gamma - \sigma_i$ (recall $\sigma_i = |\mathcal{O}_i|$, and γ is the number of distinct eigenvalues of $\mathbf{A}$). Define $\mathbf{B}_j = \begin{bmatrix} \mathbf{0} \cdots \mathbf{I}_{o_j} \cdots \mathbf{0} \end{bmatrix}$ as the matrix which extracts the j-th sub-state from the state vector $\mathbf{z}[k]$, i.e., we have $\mathbf{z}^{(j)}[k] = \mathbf{B}_j \mathbf{z}[k]$. Also, let $\mathbf{w}_{il} = \begin{bmatrix} w_{il}^{n_1} \cdots w_{il}^{n_{\gamma_i}} \end{bmatrix}^T$ denote the vector of consensus weights the i-th node assigns to the l-th node ($l \in \mathcal{N}_i$) for the estimation of the sub-states corresponding to each of its undetectable eigenvalues. Then, noting the definition of $\bar{\mathbf{J}}_{\mathcal{UO}_i}$ and using the consensus equation given by (2.35), we obtain

$$\hat{\mathbf{z}}_{\mathcal{UO}_i}[k+1] = \bar{\mathbf{J}}_{\mathcal{UO}_i} \mathbb{B}_i \sum_{l \in \mathcal{N}_i} \mathbf{w}_{il} \otimes \hat{\mathbf{z}}_l[k], \tag{2.37}$$

where $\mathbb{B}_i = diag(\mathbf{B}_{n_1}, \cdots, \mathbf{B}_{\gamma_i})$. Noting that $\mathbf{x}[k] = \mathcal{T}\mathbf{z}[k]$, and $\mathbf{z}[k] = \mathbb{P}_i\bar{\mathbf{z}}_i[k]$ (recall $\bar{\mathbf{z}}_i[k] = \begin{bmatrix}\mathbf{z}_{\mathcal{O}_i}[k]^T & \mathbf{z}_{\mathcal{UO}_i}[k]^T\end{bmatrix}^T$), and using equations (2.34) and (2.37), we obtain the governing equations of the distributed observer maintained at node i as

$$\boxed{\begin{aligned}
\hat{\mathbf{s}}_i[k+1] &= \mathbb{J}_i\hat{\mathbf{s}}_i[k] + \mathbb{L}_i(\mathbf{y}_i[k] - \mathbb{F}_i\hat{\mathbf{s}}_i[k]), && \text{(2.38a)}\\
\hat{\mathbf{z}}_{\mathcal{O}_i}[k+1] &= \begin{bmatrix}\mathbf{I}_{o_i} & \mathbf{0}\end{bmatrix}\hat{\mathbf{s}}_i[k+1], && \text{(2.38b)}\\
\hat{\mathbf{z}}_{\mathcal{UO}_i}[k+1] &= \bar{\mathbf{J}}_{\mathcal{UO}_i}\mathbb{B}_i\sum_{l\in\mathcal{N}_i}\mathbf{w}_{il}\otimes(\mathcal{T}^{-1}\hat{\mathbf{x}}_l[k]), && \text{(2.38c)}\\
\hat{\mathbf{x}}_i[k+1] &= \mathcal{T}\mathbb{P}_i\begin{bmatrix}\hat{\mathbf{z}}_{\mathcal{O}_i}[k+1]\\ \hat{\mathbf{z}}_{\mathcal{UO}_i}[k+1]\end{bmatrix}. && \text{(2.38d)}
\end{aligned}}$$

Note that since $\mathcal{T}$ depends only on the system matrix $\mathbf{A}$ which is assumed to be time-invariant, the term $\mathcal{T}^{-1}$ appearing in (2.38c) needs to be computed only once.

2.6.2 Analysis of the Estimation Scheme for Systems and Graphs Satisfying Condition 2

The following is the main result of this section.

Theorem 2.6.2 *Consider a system* $(\mathbf{A}, \mathbf{C})$ *and graph* $\mathcal{G}$ *satisfying Condition* 2*. Then, for each node* $i \in \{1, 2, \cdots, N\}$*, there exists a choice of observer gain matrix* $\mathbb{L}_i$*, and consensus weights* $w_{il}^{(j)}$*,* $\forall\lambda_j \in \mathcal{UO}_i$*,* $l \in \mathcal{N}_i$*, such that the update rules given by (2.38) form a distributed observer.*

Proof Let a system $(\mathbf{A}, \mathbf{C})$ and graph $\mathcal{G}$ satisfy Condition 2. Consider $\lambda_j \in \Lambda_U(\mathbf{A})$. Let $\mathcal{S}_j = \{m_1, \cdots, m_{\tau_j}\}$ be the set of root nodes for eigenvalue λ_j, where $\tau_j = |\mathcal{S}_j|$. Define $\mathbf{e}_{m_i}^{(j)}[k] \triangleq \hat{\mathbf{z}}_{m_i}^{(j)}[k] - \mathbf{z}^{(j)}[k]$ as the error in estimation of the j-th sub-state by the m_i-th node. The errors in estimation of $\mathbf{z}^{(j)}[k]$ for the nodes that can detect λ_j are stacked into the composite error vector $\mathbf{E}_{\mathcal{O}}^{(j)}[k]$, defined as

$$\mathbf{E}_{\mathcal{O}}^{(j)}[k] \triangleq \begin{bmatrix}\mathbf{e}_{m_1}^{(j)}[k]\\ \vdots \\ \mathbf{e}_{m_{\tau_j}}^{(j)}[k]\end{bmatrix}. \tag{2.39}$$

Similarly, we stack the estimation errors of $\mathbf{z}^{(j)}[k]$ for the nodes that cannot detect λ_j into the composite error vector $\mathbf{E}^{(j)}_{\mathcal{UO}}[k]$, defined as

$$\mathbf{E}^{(j)}_{\mathcal{UO}}[k] \triangleq \begin{bmatrix} \mathbf{e}^{(j)}_{m_{\tau_j+1}}[k] \\ \vdots \\ \mathbf{e}^{(j)}_{m_N}[k] \end{bmatrix}, \tag{2.40}$$

where $\mathcal{V} \setminus \mathcal{S}_j = \{m_{\tau_j+1}, \cdots, m_N\}$ represents a topological ordering of the non-root nodes consistent with a set of directed trees rooted at $\mathcal{S}_j$, which span $\mathcal{V} \setminus \mathcal{S}_j$. Such a set of trees exists based on Condition 2. Noting from (2.28) that $\mathbf{z}^{(j)}[k+1] = \mathbf{J}_j \mathbf{z}^{(j)}[k]$, and using equation (2.35), for $\lambda_j \in \mathcal{UO}_i$, the estimation error for the j-th sub-state by the i-th node is

$$\mathbf{e}^{(j)}_i[k+1] = \mathbf{J}_j \sum_{l \in \mathcal{N}_i} w^{(j)}_{il} \mathbf{e}^{(j)}_l[k]. \tag{2.41}$$

From (2.41), it follows that the relation between $\mathbf{E}^{(j)}_{\mathcal{O}}[k]$ and $\mathbf{E}^{(j)}_{\mathcal{UO}}[k]$ can be expressed via the equation

$$\begin{aligned} \mathbf{E}^{(j)}_{\mathcal{UO}}[k+1] &= \left(\begin{bmatrix} \mathbf{W}^j_{11} & \mathbf{W}^j_{12} \end{bmatrix} \otimes \mathbf{J}_j \right) \begin{bmatrix} \mathbf{E}^{(j)}_{\mathcal{O}}[k] \\ \mathbf{E}^{(j)}_{\mathcal{UO}}[k] \end{bmatrix} \\ &= \left(\mathbf{W}^j_{12} \otimes \mathbf{J}_j \right) \mathbf{E}^{(j)}_{\mathcal{UO}}[k] + \left(\mathbf{W}^j_{11} \otimes \mathbf{J}_j \right) \mathbf{E}^{(j)}_{\mathcal{O}}[k], \end{aligned} \tag{2.42}$$

where the weight matrix $\mathbf{W}^j = \begin{bmatrix} \mathbf{W}^j_{11} & \mathbf{W}^j_{12} \end{bmatrix}$ contains weights based on equation (2.41) (note that $\mathbf{W}^j \in \mathbb{R}^{(N-\tau_j)\times N}$, and $\mathbf{W}^j_{11}$ represents the first τ_j columns of $\mathbf{W}^j$, where $\tau_j = |\mathcal{S}_j|$). Using the same design philosophy for the consensus weights as in Condition 1, we assign non-zero consensus weights only along the branches of the spanning forest rooted at $\mathcal{S}_j$. In this way, $\mathbf{W}^j_{12}$ can be made lower triangular with zero eigenvalues (without violating the stochasticity condition imposed by equation (2.36)). We conclude that by an appropriate choice of weights as described above, we can achieve $\Lambda_U(\mathbf{W}^j_{12} \otimes \mathbf{J}_j) = \emptyset$ (even though $\lambda_j \in \Lambda_U(\mathbf{A})$).

Based on Lemma 2.6.1, each node $i \in \mathcal{V}$ can locally design its observer gain $\mathbb{L}_i$ to stabilize the local Luenberger observer error dynamics. Specifically, the error dynamics corresponding to the estimation of $\mathbf{z}^{(j)}[k]$ for each root node in $\mathcal{S}_j$ is guaranteed to asymptotically converge based on Lemma 2.6.1, i.e., the composite error $\mathbf{E}^{(j)}_{\mathcal{O}}[k]$ asymptotically converges to zero. Using ISS, we infer from (2.42) that $\lim_{k\to\infty}\mathbf{E}^{(j)}_{\mathcal{UO}}[k] = \mathbf{0}$. The same argument holds $\forall \lambda_j \in \Lambda_U(\mathbf{A})$. Thus, we conclude that every node can asymptotically estimate the sub-states of $\mathbf{z}[k]$ corresponding to both its detectable and undetectable eigenvalues, i.e., it can estimate the entire transformed state $\mathbf{z}[k]$ asymptotically. As $\mathbf{x}[k] = \mathcal{T}\mathbf{z}[k]$, each node can asymptotically estimate the true state $\mathbf{x}[k]$ as well. ■

Remark 2.6.3 *Based on the distributed observer given by (2.38), note that the dimension of the observer is equal to the sum of the dimensions of the vectors $\hat{\mathbf{s}}_i[k]$ and $\hat{\mathbf{z}}_{\mathcal{UO}_i}[k]$, and can be higher than the dimension of the state $\mathbf{x}[k]$ (as $\hat{\mathbf{s}}_i[k]$ can have a dimension higher than $\hat{\mathbf{z}}_{\mathcal{O}_i}[k]$). This augmentation is a consequence of the fact that at present, although we are able to estimate the portion $\mathbf{w}_{\mathcal{O}_i}[k]$ of the vector $\mathbf{z}_{\mathcal{UO}_i}[k]$ via the local Luenberger observer maintained at node i (given by (2.34)), we use this information only for updating $\hat{\mathbf{z}}_{\mathcal{O}_i}[k]$, and rely on consensus for estimating the entire vector $\hat{\mathbf{z}}_{\mathcal{UO}_i}[k]$ (via equation (2.38c)). This redundancy in information may be potentially overcome using a more complicated scheme where one uses consensus for estimating only the portion of the state corresponding to the vector $\mathbf{w}_{\mathcal{UO}_i}[k]$ in equation (2.31); to avoid cluttering the exposition, we omit further investigation of this issue here. However, for certain special cases of Condition 2 where the system matrix has more structure, it may be possible to construct distributed observers without state augmentation, using the approach proposed for Condition 2. For example, if the system has distinct eigenvalues, then the matrix $\bar{\mathbf{C}}_{\mathcal{UO}_i}$ in (2.29) will be zero $\forall i \in \mathcal{V}$, thereby precluding the need for state augmentation.*

2.6.3 Summary of the Estimation Scheme for Systems and Graphs Satisfying Condition 2

Similar to the strategy adopted for systems and graphs satisfying Condition 1, the distributed observer design for systems and graphs satisfying Condition 2 also constitutes an initialization (or design) phase which needs to be implemented just once, followed up by an estimation phase. However, the extra structure provided by Condition 2 allows each of these phases to be implemented in a distributed manner. The main steps of the overall scheme are summarized as follows:

Design Phase:

- All nodes simultaneously perform a common co-ordinate transformation, which brings the state matrix $\mathbf{A}$ into its Jordan canonical form. Using this form, each node identifies its locally detectable and undetectable eigenvalues.
- For each $\lambda_j \in \Lambda_U(\mathbf{A})$, the nodes run a distributed algorithm (we shall discuss such an algorithm shortly) to construct trees with roots in $\mathcal{S}_j$, which span $\mathcal{V}\setminus\mathcal{S}_j$. These trees guide the design of the consensus weight matrices to be used for each unstable and marginally stable eigenvalue of the system.

Estimation Phase:

- Each node uses a Luenberger observer for estimating the sub-states corresponding to the detectable eigenvalues, and runs consensus dynamics for estimating the sub-states corresponding to the undetectable eigenvalues. These dynamics are captured by (2.38).

Construction of Spanning Trees for Consensus Weight Design

To construct directed trees rooted at nodes in $\mathcal{S}_j$, which span $\mathcal{V} \setminus \mathcal{S}_j$, for each $\lambda_j \in \Lambda_U(\mathbf{A})$, (these trees in turn guide the construction of the consensus weight matrices) one can use standard distributed tree construction algorithms (such as Breadth-First Search (BFS)) [62]. The essential idea behind such algorithms is that each desired root node broadcasts a message indicating that it is a root, which is then passed through the network. When a node first receives such a message from a neighbor, it adopts that neighbor as its parent in the tree and rebroadcasts the message. At the conclusion of the algorithm, all nodes are aware of their parent in their tree (as long as there is a path from the root node(s) to all other nodes). For our purpose, such a distributed algorithm can be implemented by the nodes for each $\lambda_j \in \Lambda_U(\mathbf{A})$, with $\mathcal{S}_j$ representing the roots of the tree. In this way, for each $\lambda_j \in \Lambda_U(\mathbf{A})$, a node in $\mathcal{V} \setminus \mathcal{S}_j$ will identify its parent node in one of the directed trees rooted at $\mathcal{S}_j$, and as discussed in the proof of Theorem 2.6.2, will assign a non-zero consensus weight to only this parent node for the estimation of $\mathbf{z}^{(j)}[k]$.

Note that the simpler distributed observer scheme developed for systems and graphs satisfying Condition 2 may not always be applicable to systems and graphs satisfying Condition 1. To see this, consider the system and graph given by equation (2.3) and Figure 2.1, which satisfies Condition 1 but not Condition 2. As pointed out in Remark 2.3.1, the only root node that can detect the unstable eigenvalue $\lambda = 2$ belongs to the source component comprised of the isolated node 3. To implement the scheme developed for systems and graphs satisfying Condition 2, one needs to construct a tree rooted at node 3 which spans nodes 1 and 2. This is clearly not possible for this particular network; hence the method developed for Condition 2 is not applicable to this system and graph. In this case, the general distributed observer framework developed for systems and graphs satisfying Condition 1 would still apply, however.

2.7 Robustness to Communication Losses

In this section, we discuss how the general framework for distributed observer design that we have described thus far (the idea of using Luenberger observers for estimating the locally detectable states and consensus dynamics for the remaining states) can be extended to account for time-varying communication graphs that are a consequence of communication link failures. We only perform the analysis for systems and networks satisfying Condition 1, since similar arguments will hold for Condition 2 as well. We consider a scenario where the network varies with time due to failure or recovery of subsets of edges of the baseline graph $\mathcal{G}$. We denote this class of switching signals by Ω. Under the class of switching signals Ω, the time-varying communication graph is denoted by $\mathcal{G}_{\sigma(k)} = (\mathcal{V}, \mathcal{E}_{\sigma(k)})$, where $\sigma(k)$ is a finite index set representing the different switching modes, and $\mathcal{E}_{\sigma(k)} \subseteq \mathcal{E}$ (recall that $\mathcal{E}$ represents the set of edges of the baseline graph $\mathcal{G}$). For the rest of the analysis in this section, we assume that the baseline communication graph $\mathcal{G}$ is strongly-connected, i.e., $\mathcal{G}$ is strongly-connected in the absence of link failures.

We make two minor modifications to our original estimation strategy (refer to Section 2.5.6) to account for communication losses. First, during the design phase, for estimation of a given sub-state, we construct a spanning directed acyclic graph (DAG) (instead of a spanning tree) rooted at the corresponding source node to allow for the possibility of having redundant communication links. Accordingly, in the DAG constructed for estimation of sub-state j (where $j \in \{1, \cdots, N\}$), let the set of parent nodes for node i be denoted by $\mathcal{P}_i^{(j)}$. Second, for estimating the j-th sub-state, where $j \in \{1, \cdots, N\} \setminus \{i\}$, the i-th node does the following: if at a certain time-step it receives information from only a proper subset of its parent set $\mathcal{P}_i^{(j)}$, then it still employs equation (2.11), redistributing the weights among such a subset so as to preserve the stochasticity constraint imposed by (2.12). For the more critical scenario where node i gets disconnected from all its parents in the set $\mathcal{P}_i^{(j)}$ at a given time-step k, it updates $\hat{\mathbf{z}}_i^{(j)}[k]$ using previous values of its *own* estimates in the following way:

$$\hat{\mathbf{z}}_i^{(j)}[k+1] = \mathbf{A}_{jj}\hat{\mathbf{z}}_i^{(j)}[k] + \sum_{l=1}^{j-1} \mathbf{A}_{jl}\hat{\mathbf{z}}_i^{(l)}[k]. \tag{2.43}$$

A direct consequence of the update rule (2.43) is that the matrix $\mathbf{A}_{jj}$, which may be unstable, appears in the block diagonal position corresponding to node i in the lower block triangular error dynamics matrix $\mathbf{M}_j$ given by (2.22). Notice also that for a given node i, the observer update equations (2.10) and (2.13) are unaffected by changes in the network structure. To proceed with our analysis, we make the following assumption on the class of switching signals Ω.

Assumption 2.7.1 *The class of switching signals Ω has the following property: there exists a positive integer T such that in every time interval of the form $[kT, (k+1)T)$,* *where $k \in \mathbb{N}$, for each sub-state $j \in \{1, \cdots, N\}$, for every node $i \in \mathcal{V} \setminus \{j\}$, there exists an integer $l \in [kT, (k+1)T)$ such that $\mathcal{G}_{\sigma(l)}$ contains an edge from at least one node in $\mathcal{P}_i^{(j)}$ to node i.*[11]

In words, Assumption 2.7.1 simply implies that within each interval $[kT, (k+1)T)$, for each sub-state j, every non-source node $\mathcal{V} \setminus \{j\}$ is guaranteed to receive information from at least one of its parents in $\mathcal{P}_i^{(j)}$ at least once over the entire interval. With this in mind, we now state the main result of this section.

Theorem 2.7.2 *Consider a system $(\mathbf{A}, \mathbf{C})$ and a strongly-connected baseline communication graph $\mathcal{G}$ satisfying Condition 1. Suppose the class of switching signals Ω satisfies Assumption 2.7.1. Then, equations (2.10), (2.11) with time-varying weights, (2.13) and* (2.43) *form a distributed observer.*

The proof is provided in Section 2.11.2.

Remark 2.7.3 *The conditions in Theorem 2.7.2 allude to the preservation of certain information flow patterns over contiguous, non-overlapping bounded time intervals and in the process inform the judicious placement of redundant communications links for augmenting network robustness against communication failures. If such conditions are met, then the strategy described in this section can be used to deal with intermittent communication losses without the need for a redesign of the estimation scheme on the fly. However, such a redesign cannot be avoided if one of the sensors involved in the multi-sensor observable canonical decomposition described in Section 2.5.1 fails. This is a limitation of the scheme proposed for systems and graphs satisfying Condition 1; such a limitation is likely to be a shortcoming of the existing distributed observer constructions [1, 27–32] as well, since they typically involve a centralized design phase and do not account for node or link failures.*

Remark 2.7.4 *While the foregoing discussion focused on transient communication failures, permanent communication failures can also be accommodated within our*

[11] Note that within a given interval of the form $[kT, (k+1)T)$, for a specific sub-state j, l might be different for the different non-source nodes in $\mathcal{V} \setminus \{j\}$.

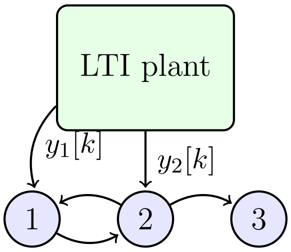


Fig. 2.4. Network topology for the example system considered in Section 2.8.

framework in the following way. Suppose a non-source node i for a certain sub-state gets disconnected from all its parent nodes in the baseline graph $\mathcal{G}$ at a certain time-step. If $\mathcal{G}$ is undirected and remains connected after such faults, then node i can broadcast the fault status back to the corresponding source node, thereby initiating the construction of a new tree that retains information flow from the source to node i. Such trees can easily be constructed in a distributed manner using standard techniques [62] similar to the one outlined earlier in Section 2.6.3. The non-source nodes can then re-adjust their consensus weights based on the new tree and employ (2.11) *as earlier. Note that the above strategy would also apply to permanent sensor failures provided such sensors are not involved in the multi-sensor observable canonical decomposition.*[12]

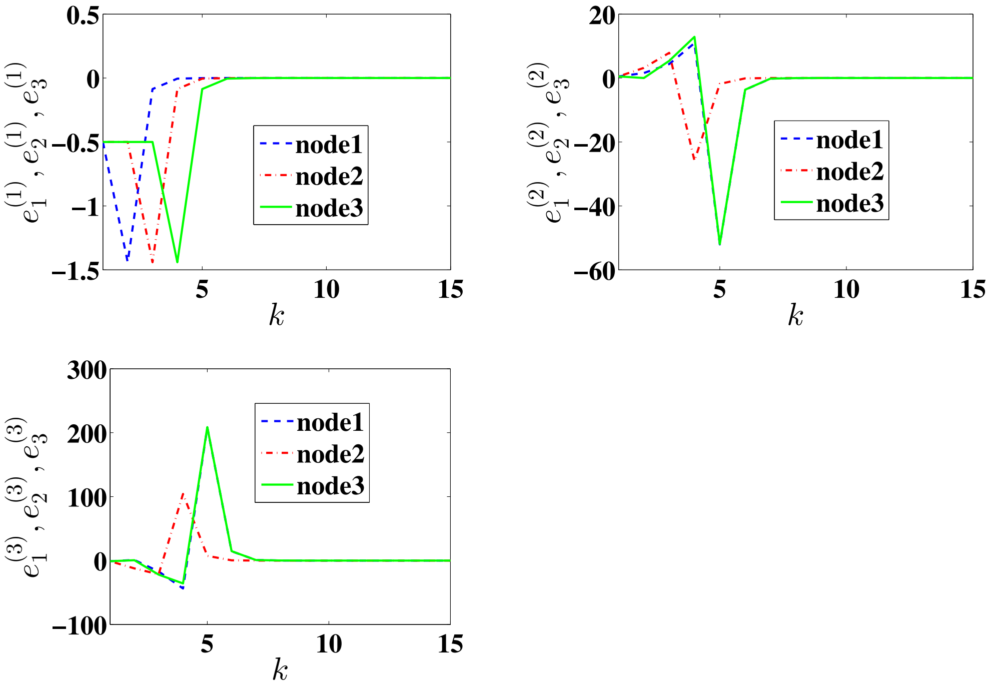


Fig. 2.5. *(Top left)* Error dynamics for the first state. *(Top right)* Error dynamics for the second state. *(Bottom left)* Error dynamics for the third state.

2.8 Example

In this section, we present an example to illustrate the scheme developed for Condition 1. To this end, consider the network shown in Figure 2.4, and the associated system and measurement matrices given by

$$\mathbf{A} = \begin{bmatrix} 1 & 0 & 0 \\ 2 & 2 & 0 \\ -5 & 0 & 2 \end{bmatrix}, \mathbf{C}_1 = \begin{bmatrix} 4 & 4 & 1 \end{bmatrix}, \mathbf{C}_2 = \begin{bmatrix} 11 & 13 & 3 \\ 16 & 18 & 4 \end{bmatrix}, \mathbf{C}_3 = 0. \tag{2.44}$$

Note that the system is not detectable from any individual node. It can be verified that the pair $(\mathbf{A}, \begin{bmatrix} \mathbf{C}_1 \\ \mathbf{C}_2 \end{bmatrix})$, associated with the source component comprised of nodes 1 and 2, is observable. Hence, the scheme developed for Condition 1 is applicable to this setting. To implement the multi-sensor observable canonical decomposition, we start by bringing the pair $(\mathbf{A}, \mathbf{C}_1)$ to the observable canonical form. This is achieved using

$$\mathbf{T}_1 = \begin{bmatrix} 4 & 7 & 0 \\ 4 & 8 & -0.2425 \\ 1 & 2 & 0.9701 \end{bmatrix}. \tag{2.45}$$

Since $(\mathbf{A}, \begin{bmatrix} \mathbf{C}_1 \\ \mathbf{C}_2 \end{bmatrix})$ is observable, in this specific case, we have $\mathbf{A}_{22} = \mathbf{A}_{1\mathcal{U}}$ and $\mathbf{A}_{2\mathcal{U}} = 0$ (here we use notations consistent with the ones described in the proof of Proposition 2.5.1 in Section 2.11.1). Thus, we have $\mathbf{T}_2 = \mathbf{I}_3$ and $\mathcal{T} = \mathbf{T}_1\mathbf{T}_2 = \mathbf{T}_1$. Using $\mathcal{T}$, we perform the multi-sensor observable canonical decomposition to obtain

$$\bar{\mathbf{A}} = \left[\begin{array}{cc|c} -10.9412 & -22.6471 & 0 \\ 6.8235 & 13.9412 & 0 \\ \hline -21.3431 & -37.3505 & 2 \end{array}\right], \bar{\mathbf{C}}_1 = \left[\begin{array}{cc|c} 33 & 62 & 0 \end{array}\right], \bar{\mathbf{C}}_2 = \left[\begin{array}{cc|c} 99 & 187 & -0.2425 \\ 140 & 264 & -0.4851 \end{array}\right]. \tag{2.46}$$

[12] Clearly, for an undirected baseline graph that is initially k-connected, $k-1$ permanent communication faults can be accounted for in the aforementioned manner. In fact, $k-1$ permanent sensor failures could also be handled if such sensors are not involved in the multi-sensor observable decomposition.

Based on the above decomposition, and the theory developed for Condition 1, it is easy to see that the transformed state $\mathbf{z}[k] = \mathcal{T}^{-1}\mathbf{x}[k]$ will contain two sub-states and the unobservable sub-state will be of zero-dimension. The local Luenberger observer gains are chosen as $\mathbf{L}_1 = \begin{bmatrix} -4.6404 & 2.5174 \end{bmatrix}^T$ and $\mathbf{L}_2 = \begin{bmatrix} -1.641 & -3.282 \end{bmatrix}$. Noting that node 1 is responsible for estimating sub-state 1, and node 2 for sub-state 2, the consensus weight vectors used by the two nodes are $\mathbf{w}_{11} = \begin{bmatrix} 1 & 0 \end{bmatrix}^T, \mathbf{w}_{12} = \begin{bmatrix} 0 & 1 \end{bmatrix}^T, \mathbf{w}_{21} = \begin{bmatrix} 1 & 0 \end{bmatrix}^T$, and $\mathbf{w}_{22} = \begin{bmatrix} 0 & 1 \end{bmatrix}^T$. Section 2.5.5 provides a description of these weight vectors. Using these design parameters, nodes 1 and 2 maintain the following estimators of the form (2.25):

$$\hat{\mathbf{x}}_1[k+1] = \mathbf{N}\hat{\mathbf{x}}_1[k] + \mathcal{T}\mathbb{H}_1\left(\mathbf{y}_1[k] - \mathbf{C}_1\hat{\mathbf{x}}_1[k]\right) + \mathbb{G}_{11}\hat{\mathbf{x}}_1[k] + \mathbb{G}_{12}\hat{\mathbf{x}}_2[k],$$
$$\hat{\mathbf{x}}_2[k+1] = \mathbf{N}\hat{\mathbf{x}}_2[k] + \mathcal{T}\mathbb{H}_2\left(\mathbf{y}_2[k] - \mathbf{C}_2\hat{\mathbf{x}}_2[k]\right) + \mathbb{G}_{21}\hat{\mathbf{x}}_1[k] + \mathbb{G}_{22}\hat{\mathbf{x}}_2[k],$$

where

$$\mathbf{N} = \begin{bmatrix} 0 & 0 & 0 \\ 1.29 & 0 & 0 \\ -5.18 & 0 & 0 \end{bmatrix}, \mathcal{T}\mathbb{H}_1 = \begin{bmatrix} -0.94 \\ 1.58 \\ 0.39 \end{bmatrix}, \mathcal{T}\mathbb{H}_2 = \begin{bmatrix} 0 & 0 \\ 0.40 & 0.80 \\ -1.59 & -3.18 \end{bmatrix},$$

$$\mathbb{G}_{11} = \mathbb{G}_{21} = \begin{bmatrix} 1 & 0 & 0 \\ 0.71 & 1.88 & 0.47 \\ 0.18 & 0.47 & 0.12 \end{bmatrix}, \mathbb{G}_{12} = \mathbb{G}_{22} = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0.12 & -0.47 \\ 0 & -0.47 & 1.88 \end{bmatrix}.$$

Since node 3 does not belong to any source component, it simply runs a pure consensus strategy given by $\hat{\mathbf{x}}_3[k+1] = \mathbf{A}\hat{\mathbf{x}}_2[k]$. Note that all nodes maintain observers of dimension 3. For simulations, $\mathbf{x}[0] = \begin{bmatrix} 0.5 & -0.5 & 1 \end{bmatrix}^T$, and the initial estimates of all three nodes are set to $\mathbf{0}$. Figure 2.5 shows the evolution of the estimation errors (in these plots, the notation $e_i^{(j)}$ is used to denote the error in estimation of state j by node i) and validates the scheme developed for Condition 1.

2.9 Distributed Functional Observers

Consider a scenario where each node in the network is interested in tracking the evolution of the functional $\mathbf{1}_n^T\mathbf{x}[k]$, i.e., each node is interested in estimating the sum of the states of the system. Of course, this can trivially be achieved by first estimating the entire state vector and then computing the sum of its components. However, given the modest objective, this approach can be computationally prohibitive for systems with several states. The above discussion leads to the following question: Is it possible to reduce the online computations per node based on an alternate approach? Answering this question formally is the subject of this section. To this end, suppose the goal of each node is to estimate $\boldsymbol{\psi}[k]$, where

$$\boldsymbol{\psi}[k] = \mathbf{L}\mathbf{x}[k]. \tag{2.47}$$

Here, $\mathbf{L} \in \mathbb{R}^{r \times n}$ is a full row-rank matrix (without loss of generality); hence $\boldsymbol{\psi}[k]$ represents r linearly independent functionals of the state.[13] While there is a rich body of literature that looks at the centralized functional observer design problem (see [63, 64]), we are unaware of any work that investigates the distributed counterpart based on the model considered here. In [65], the authors develop a partially distributed functional observer scheme for coupled interconnected LTI systems, where each sub-system maintains an observer for estimating functionals of the state corresponding to that particular sub-system. However, the model and problem formulation in [65] differs from the one considered in this chapter. A key point of difference is that in [65], the impact of the underlying communication graph does not play a role in the design strategy, whereas our approach focuses on analyzing the interplay between the system dynamics and the network topology.

The main contribution of this section is a distributed algorithm that guarantees asymptotic reconstruction of $\boldsymbol{\psi}[k]$ at each sensor node, under certain conditions on the system dynamics and network topology.

[13] When $\mathbf{L}$ is the identity matrix (or more generally a square non-singular matrix), we recover the distributed state estimation problem studied earlier in this chapter.

2.9.1 Problem Formulation

Consider the LTI system given by (2.1), the measurement model specified by (2.2), the functionals of interest described by (2.47), and a predefined directed communication graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ where $\mathcal{V}$ represents the set of N nodes (or sensors). Let $\hat{\boldsymbol{\psi}}_i[k]$ denote the estimate of $\boldsymbol{\psi}[k]$ maintained by node i. Given this setting, the problem studied in this section is formally stated as follows.

Problem 2 (Distributed Functional Estimation Problem) *For the model specified by equations* (2.1)*,* (2.2)*,* (2.47) *and a predefined communication graph* $\mathcal{G}$*, design a distributed algorithm that achieves* $\lim_{k\to\infty} \|\hat{\boldsymbol{\psi}}_i[k] - \boldsymbol{\psi}[k]\| = 0, \forall i \in \{1, \cdots, N\}$.

Remark 2.9.1 *As long the system is globally detectable, i.e., the pair* $(\mathbf{A}, \mathbf{C})$ *is detectable, and the communication graph* $\mathcal{G}$ *is strongly connected, a trivial way to solve Problem 2 is to reconstruct the entire state* $\mathbf{x}[k]$ *at every sensor node based on any of the existing distributed state estimation techniques in [1, 30, 58]. Our goal in this section will be to design observers that are in general of order*[14] *smaller than the dimension of the state* $\mathbf{x}[k]$*. For more on this issue, refer to Remark 2.9.7.*

A distributed algorithm that solves the above problem will be called a *distributed functional observer*. Note that in general it may not be possible for a given node i to estimate $\boldsymbol{\psi}[k]$ by solely relying on its own measurements,[15] thereby dictating the need to exchange information with its neighbors. In the next section, we show that existing results/techniques for the centralized version of the problem are not directly applicable to the problem under consideration.

[14] By the order of an observer at a given sensor node, we refer to the dimension of the portion of the state that is dynamically estimated at that node, i.e., the portion that is not obtained directly from the measurements of the node under consideration.

[15] This is precisely the case when $\boldsymbol{\psi}[k]$ is a linear function of some states that are undetectable with respect to the measurements of node i; for related notions of 'Functional Observability', see [66, 67].

2.9.2 Motivation

The purpose of this section is to illustrate that classical existence conditions for the design of centralized functional observers (of a given order) do not generally hold in a distributed setting, and in the process, motivate our present work. To this end, we first recall the following necessary and sufficient conditions set forth by Darouach [63] for the existence of a centralized functional observer of order r, where $r = \text{rank}\,\mathbf{L}$:

(i)

$$\text{rank}\begin{bmatrix}\mathbf{LA}\\ \mathbf{CA}\\ \mathbf{C}\\ \mathbf{L}\end{bmatrix} = \text{rank}\begin{bmatrix}\mathbf{CA}\\ \mathbf{C}\\ \mathbf{L}\end{bmatrix}, \tag{2.48}$$

(ii)

$$\text{rank}\begin{bmatrix}s\mathbf{L}-\mathbf{LA}\\ \mathbf{CA}\\ \mathbf{C}\end{bmatrix} = \text{rank}\begin{bmatrix}\mathbf{CA}\\ \mathbf{C}\\ \mathbf{L}\end{bmatrix}, \ \forall s \in \mathbb{C}, \ \ |s| \geq 1. \tag{2.49}$$

Next, consider the following model, where the system is monitored by a network of nodes, as depicted by Figure 2.6:

$$\begin{gathered}\underbrace{\begin{bmatrix}x^{(1)}[k+1]\\ x^{(2)}[k+1]\end{bmatrix}}_{\mathbf{x}[k+1]} = \underbrace{\begin{bmatrix}\frac{1}{2} & 2\\ 0 & 3\end{bmatrix}}_{\mathbf{A}}\underbrace{\begin{bmatrix}x^{(1)}[k]\\ x^{(2)}[k]\end{bmatrix}}_{\mathbf{x}[k]},\\ \mathbf{L} = \begin{bmatrix}1 & 0\end{bmatrix}, \mathbf{C}_1 = \begin{bmatrix}0 & 1\end{bmatrix}, \mathbf{C}_2 = \mathbf{C}_3 = 0.\end{gathered} \tag{2.50}$$

The objective is to asymptotically estimate $x^{(1)}[k]$ at each of the three sensor nodes. It is easy to verify that the necessary and sufficient conditions (equations (2.48) and (2.49)) for the existence of a centralized 1st order functional observer are satisfied by the model (2.50) with $\mathbf{C} = \begin{bmatrix}\mathbf{C}_1^T & \mathbf{C}_2^T & \mathbf{C}_3^T\end{bmatrix}^T$. At this point, the natural inclination is to ascertain whether it is possible for each sensor node to asymptotically estimate $x^{(1)}[k]$ via 1st order estimators. To formally answer this question, we need to impart

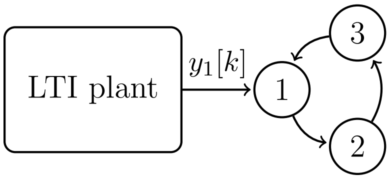


Fig. 2.6. Example for illustrating Proposition 2.9.1.

some structure to the distributed observers under consideration. To this end, consider distributed observers of the form[16]

$$\hat{x}_i^{(1)}[k+1] = \alpha_i \sum_{j \in \mathcal{N}_i} w_{ij} \hat{x}_j^{(1)}[k] + \beta_i \sum_{j \in \mathcal{N}_i} y_j[k], \tag{2.51}$$

where $\hat{x}_i^{(1)}[k]$ is the estimate of state $x^{(1)}[k]$ maintained by node i at time-step k, and α_i, β_i, w_{ij} are free design parameters at node i. Moreover, the weights w_{ij} are non-negative and satisfy $\sum_{j \in \mathcal{N}_i} w_{ij} = 1, i \in \{1,2,3\}$. We have the following simple result.

Proposition 2.9.1 *For the model given by* (2.50)*, and the corresponding network depicted by Figure 2.6, it is impossible for node* 3 *to estimate the function* $\mathbf{L}\mathbf{x}[k] = x^{(1)}[k]$ *using a 1st order observer that has structure given by* (2.51)*.*

Proof Let $e_i[k] = \hat{x}_i^{(1)}[k] - x^{(1)}[k]$ denote the error in estimation of state $x^{(1)}[k]$ at node i. Based on (2.50), we have $x^{(1)}[k+1] = \alpha x^{(1)}[k] + \beta y_1[k]$ where $\alpha = \frac{1}{2}$, $\beta = 2$

[16] The choice of this observer structure is inspired by the fact that standard distributed state observers existing in literature are essentially of this form.

and $y_1[k] = x^{(2)}[k]$. Then, using (2.51) and some straightforward algebra, we obtain the following error dynamics:

$$\underbrace{\begin{bmatrix} e_1[k+1] \\ e_2[k+1] \\ e_3[k+1] \end{bmatrix}}_{\mathbf{e}[k+1]} = \underbrace{\begin{bmatrix} \alpha_1 w_{11} & 0 & \alpha_1 w_{13} \\ \alpha_2 w_{21} & \alpha_2 w_{22} & 0 \\ 0 & \alpha_3 w_{32} & \alpha_3 w_{33} \end{bmatrix}}_{\mathbf{M}} \underbrace{\begin{bmatrix} e_1[k] \\ e_2[k] \\ e_3[k] \end{bmatrix}}_{\mathbf{e}[k]} + \underbrace{\begin{bmatrix} (\alpha_1 - \alpha) \\ (\alpha_2 - \alpha) \\ (\alpha_3 - \alpha) \end{bmatrix}}_{\mathbf{B}_1} x^{(1)}[k] + \underbrace{\begin{bmatrix} (\beta_1 - \beta) \\ (\beta_2 - \beta) \\ -\beta \end{bmatrix}}_{\mathbf{B}_2} y_1[k]. \tag{2.52}$$

To achieve $\lim_{k\to\infty} \mathbf{e}[k] = \mathbf{0}$ regardless of the initial conditions and the trajectories of $x^{(1)}[k]$ and $x^{(2)}[k]$, we require $\mathbf{M}$ to be Schur stable and $\mathbf{B}_1$ and $\mathbf{B}_2$ to be zero. To obtain $\mathbf{B}_1 = \mathbf{0}$, we must set $\alpha_1 = \alpha_2 = \alpha_3 = \alpha$. However, it is impossible to set $\mathbf{B}_2$ to zero due to the non-zero coupling term β between the functional of interest $x_1[k]$ and the measurement $y_1[k] = x_2[k]$. Based on the unstable dynamics of $y_1[k]$ (see (2.50)), it follows that no choice of free design parameters at the various nodes can guarantee $\lim_{k\to\infty} e_3[k] = \mathbf{0}$. ■

Remark 2.9.2 *For the sake of illustration, we considered the observer model given by (2.51). However, it should be noted that given the plant and measurement model (2.50), and the network depicted by Figure 2.6, adding more free design parameters to the observer structure will not change the result of Proposition 2.9.1.*

To see why Darouach's conditions do not generally hold in a distributed setting, let us take a closer look at the rank condition (2.48). We see that (2.48) implies the existence of matrices $\mathbf{M}_1, \mathbf{M}_2, \mathbf{M}_3$ (not necessarily unique) such that $\mathbf{LA} = \mathbf{M}_1\mathbf{L} + \mathbf{M}_2\mathbf{C} + \mathbf{M}_3\mathbf{CA}$. Thus, referring to equations (2.1), (2.2) and (2.47), we have $\boldsymbol{\psi}[k+1] = \mathbf{M}_1\boldsymbol{\psi}[k] + \mathbf{M}_2\mathbf{y}[k] + \mathbf{M}_3\mathbf{y}[k+1]$. Since the dynamics of $\boldsymbol{\psi}[k]$ are coupled to the measurements that are directly available in a centralized setting (and hence require no estimation), it suffices to maintain a dynamic estimator of order equal to the length of the vector $\boldsymbol{\psi}[k]$. However, in a distributed setting, the entire measurement vector $\mathbf{y}[k]$ is no longer accessible at a single sensor node, thereby precluding the direct use of a centralized functional observer design.

At this stage, it should be pointed out that even if Darouach's conditions are not met, it is still possible to construct minimal order centralized functional observers of order greater than r [64]. However, based on the discussion in this section, to isolate the challenges introduced by the distributed setting, we restrict our attention to tuples $(\mathbf{A}, \mathbf{C}, \mathbf{L})$ that allow for the construction of r-th order centralized functional observers and additionally possess certain extra structure (to be discussed later). Note that the notion of a 'minimal order distributed functional observer' is not clearly defined: one may require all nodes to maintain observers of the same order and seek to minimize such an order. Alternatively, one may allow the nodes to maintain observers of different orders and seek to minimize the average order (there might be other possible interpretations as well). In this section, we stick to the former notion and develop a distributed functional estimation strategy that requires all nodes to maintain observers of the same order that is in general smaller than the dimension of the state $\mathbf{x}[k]$. However, the problem of defining and subsequently obtaining a minimal order distributed functional observer remains open.

For the problem under consideration, since the dynamics of $\boldsymbol{\psi}[k]$ are coupled with the measurements that are no longer co-located, it becomes necessary for the sensor nodes to maintain estimates of each others' measurements in order to estimate $\boldsymbol{\psi}[k]$.[17] Building on this intuition, we introduce the notion of '*functional leader sets*' in the following section, and investigate how the dynamics of the functionals $\boldsymbol{\psi}[k]$ are coupled to the measurements of such a set of nodes via an appropriately designed similarity transformation.

[17] This is illustrated by the system considered in Proposition 2.9.1 where the functional of interest, namely $x^{(1)}[k]$, is coupled to the state measured by node 1. Hence, nodes that are not immediate neighbors of node 1 (like node 3) need to maintain estimates of $y_1[k]$ in order to estimate $x^{(1)}[k]$.

2.9.3 Functional Leader Sets

Before formally defining a functional leader set, we need to first introduce some terminology. To this end, note that by a row sub-matrix $\bar{\mathbf{C}}_{\mathcal{S}}$ of $\mathbf{C}_{\mathcal{S}}$, we imply that $\bar{\mathbf{C}}_{\mathcal{S}}$ contains a non-empty (not necessarily proper) subset of the rows of $\mathbf{C}_{\mathcal{S}}$, i.e., $\mathcal{R}(\bar{\mathbf{C}}_{\mathcal{S}}) \subseteq \mathcal{R}(\mathbf{C}_{\mathcal{S}})$. Consider the following definitions.

Definition 2.9.1 (Feasible Leader Set) *A set of nodes $\mathcal{S} \subseteq \mathcal{V}$ is called a feasible leader set if there exists at least one row sub-matrix $\bar{\mathbf{C}}_{\mathcal{S}}$ of $\mathbf{C}_{\mathcal{S}}$ satisfying the following two conditions:*

(i)

$$rank \begin{bmatrix} \mathbf{LA} \\ \bar{\mathbf{C}}_{\mathcal{S}}\mathbf{A} \\ \mathbf{L} \\ \bar{\mathbf{C}}_{\mathcal{S}} \end{bmatrix} = rank \begin{bmatrix} \mathbf{L} \\ \bar{\mathbf{C}}_{\mathcal{S}} \end{bmatrix}, \tag{2.53}$$

(ii)

$$rank \begin{bmatrix} s\begin{bmatrix} \mathbf{L} \\ \bar{\mathbf{C}}_{\mathcal{S}} \end{bmatrix} - \begin{bmatrix} \mathbf{L} \\ \bar{\mathbf{C}}_{\mathcal{S}} \end{bmatrix}\mathbf{A} \\ \bar{\mathbf{C}}_{\mathcal{S}} \end{bmatrix} = rank \begin{bmatrix} \mathbf{L} \\ \bar{\mathbf{C}}_{\mathcal{S}} \end{bmatrix}, \quad \forall s \in \mathbb{C}, \quad |s| \geq 1. \tag{2.54}$$

Definition 2.9.2 (Minimal Leader Set) *A set $\mathcal{S}$ is called a minimal leader set if $\mathcal{S}$ is a feasible leader set and no subset of $\mathcal{S}$ is a feasible leader set. A feasible leader set $\mathcal{S}$ with $|\mathcal{S}| = 1$ is considered to be minimal by default.*

Given a minimal leader set $\mathcal{S}$, if there are several row sub-matrices of $\mathbf{C}_{\mathcal{S}}$ that satisfy conditions (2.53) and (2.54), denote the row sub-matrix that produces the lowest rank of $\begin{bmatrix} \mathbf{L}^T & \bar{\mathbf{C}}_{\mathcal{S}}^T \end{bmatrix}^T$ by $\mathbf{C}_{\mathcal{S}_{min}}$ and the corresponding rank by $r_{\mathcal{S}_{min}}$. Let the set of all feasible leader sets be denoted by $\mathcal{F}$ and the set of all minimal leader sets be denoted by $\mathcal{M} = \{\mathcal{S}^{(1)}, \cdots, \mathcal{S}^{(l)}\}$, where $l = |\mathcal{M}|$. The tuples characterizing the minimal leader sets are given by $\{(\mathbf{C}_{\mathcal{S}^{(1)}_{min}}, r_{\mathcal{S}^{(1)}_{min}}), \cdots, (\mathbf{C}_{\mathcal{S}^{(l)}_{min}}, r_{\mathcal{S}^{(l)}_{min}})\}$.

Definition 2.9.3 (Functional Leader Set) *A set $\mathcal{S}^{(i)} \in \mathcal{M}$ is referred to as a functional leader set if $r_{\mathcal{S}^{(i)}_{min}} \leq r_{\mathcal{S}^{(j)}_{min}} \forall j \in \{1, \cdots, l\} \setminus \{i\}$.*

Thus, a functional leader set is a minimal leader set that yields the lowest rank on the R.H.S. of equation (2.53) among all minimal leader sets. Given any tuple $(\mathbf{A}, \mathbf{C}, \mathbf{L})$ described by (2.1), (2.2), (2.47), if $\mathcal{F}$ is non-empty, then it is easily seen that $\mathcal{M}$ is also non-empty and hence we are guaranteed the existence of at least one functional leader set. If there are multiple functional leader sets, it suffices to pick any one for our subsequent analysis since all such sets will essentially lead to distributed functional observers of the same order. Thus, if $\mathcal{F}$ is non-empty, we pick any functional leader set and denote it by $\mathcal{S}^{\star}$. For notational simplicity, we denote the tuple characterizing $\mathcal{S}^{\star}$ by $(\mathbf{C}^{\star}, r^{\star})$. The nodes in $\mathcal{S}^{\star}$ are referred to as *functional leader nodes* and it will be subsequently shown that such nodes play a key role in solving Problem 2.

Remark 2.9.3 *Roughly speaking, it will soon be apparent that any set of sensor nodes belonging to $\mathcal{F}$ (and hence $\mathcal{M}$) can effectively serve as 'leaders' in the consensus dynamics for estimating the functionals of interest, thereby justifying the proposed terminology. Furthermore, if set $\mathcal{S}^{(i)} \in \mathcal{M}$ is chosen as the leader set, then our design would result in every node maintaining a distributed functional observer of order $r_{\mathcal{S}^{(i)}_{min}}$. The definition of a functional leader set* [18] *is thus motivated by the goal of obtaining the distributed functional observer of minimal order among all feasible leader sets.*

Before proceeding further, we illustrate some of the concepts introduced in this section via the following model:

$$\mathbf{A} = \begin{bmatrix} 0 & 2 & 0 \\ 3 & 0 & 0 \\ 0 & 0 & 5 \end{bmatrix}, \mathbf{C}_1 = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}, \mathbf{L} = \begin{bmatrix} 1 & 0 & 0 \end{bmatrix}. \tag{2.55}$$

[18] Given a tuple $(\mathbf{A}, \mathbf{C}, \mathbf{L})$ described by (2.1), (2.2) and (2.47), the design of an algorithm that finds a functional leader set $\mathcal{S}^{\star}$ (provided $\mathcal{F}$ is non-empty), and the subsequent analysis of its complexity, are interesting avenues of future research.

Clearly, $\mathcal{S} = \{1\}$ is a minimal leader set with $\mathbf{C}_1$ satisfying both the rank conditions (2.53) and (2.54). However, these conditions are also satisfied by the row sub-matrix formed by considering just the first row of $\mathbf{C}_1$. While considering the entire $\mathbf{C}_1$ will lead to a distributed functional observer of order 3, considering only its first row will lead to an observer of order 2 using our design methodology. Given a minimal leader set $\mathcal{S}$, the foregoing discussion motivates the need to check whether sub-matrices of $\mathbf{C}_{\mathcal{S}}$ satisfy the conditions (2.53) and (2.54).[19] With $\mathbf{A}$ and $\mathbf{L}$ as described in (2.55), suppose we had $\mathbf{C}_1 = \begin{bmatrix} 0 & 1 & 0 \end{bmatrix}$ and $\mathbf{C}_2 = \begin{bmatrix} 0 & 0 & 1 \end{bmatrix}$. Then, $\mathcal{S} = \{1, 2\}$ would be a feasible (but not minimal) leader set, $\mathcal{S} = \{1\}$ would be a minimal leader set and $\mathcal{S} = \{2\}$ would not be a feasible leader set.

The following property of the functional leader set $\mathcal{S}^\star$ will be critical in our subsequent design.[20]

Lemma 2.9.4 *Given a tuple* $(\mathbf{A}, \mathbf{C}, \mathbf{L})$ *described by* (2.1), (2.2) *and* (2.47) *such that* $\mathcal{F}$ *is non-empty, let the functional leader set* $\mathcal{S}^\star$ *be characterized by the tuple* $(\mathbf{C}^\star, r^\star)$ *with* p *denoting the number of rows of* $\mathbf{C}^\star$*. Then, there exists a similarity transformation matrix* $\mathbf{T}$ *that brings* $(\mathbf{A}, \mathbf{C}^\star)$ *to the following form:*

$$\bar{\mathbf{A}} = \begin{bmatrix} \mathbf{A}_D & \mathbf{0} \\ \mathbf{A}_E & \mathbf{A}_F \end{bmatrix}, \; \bar{\mathbf{C}} = \begin{bmatrix} \mathbf{C}_D & \mathbf{0} \end{bmatrix}, \tag{2.56}$$

where $\mathbf{A}_D \in \mathbb{R}^{r^\star \times r^\star}$, $\mathbf{C}_D \in \mathbb{R}^{p \times r^\star}$*. Furthermore, the following properties hold: (i) the state vector corresponding to the matrix* $\mathbf{A}_D$ *has the functionals of interest* $\boldsymbol{\psi}[k]$ *as its first* r *components, and a subset of measurements corresponding to the matrix* $\mathbf{C}^\star$ *as the remaining* $r^\star - r$ *components; and (ii) the pair* $(\mathbf{A}_D, \mathbf{C}_D)$ *is detectable.*

Proof By definition, since $\mathcal{S}^\star$ is a feasible leader set, the rank conditions (2.53) and (2.54) are satisfied by $\bar{\mathbf{C}}_{\mathcal{S}^\star} = \mathbf{C}^\star$. In particular, based on the rank condition (2.53), it

[19] Intuitively, we see that the state of interest, namely state 1, is coupled only to the second state. Hence, the extra information about the third state provided by the second row of $\mathbf{C}_1$ is irrelevant in the present context. Based on this discussion, note that our approach ensures that the order of the proposed distributed functional observer is in general smaller than the dimension of the detectable subspace of the pair $(\mathbf{A}, \mathbf{C})$, where $\mathbf{C}$ represents the collective observation matrix.

[20] Clearly, any feasible leader set possesses a similar property; the rationale behind considering the functional leader set in particular is made apparent by Remark 2.9.3.

is easy to see that $\mathcal{R}(\begin{bmatrix}\mathbf{L}^T & \mathbf{C}^{\star T}\end{bmatrix}^T)$ is $\mathbf{A}^T$-invariant. Define $\mathbf{\Sigma} \triangleq \begin{bmatrix}\mathbf{L}^T & \tilde{\mathbf{C}^{\star}}^T\end{bmatrix}^T$, where $\tilde{\mathbf{C}^{\star}}$ contains all the linearly independent rows of $\mathbf{C}^{\star}$ that are also linearly independent of the rows of $\mathbf{L}$. Noting that $\operatorname{rank}\mathbf{\Sigma} = r^{\star}$, it follows that there exists a matrix $\mathbf{A}_D \in \mathbb{R}^{r^{\star}\times r^{\star}}$ such that

$$\mathbf{\Sigma A} = \mathbf{A}_D\mathbf{\Sigma}. \tag{2.57}$$

Let us define a non-singular transformation matrix as $\mathbf{T} \triangleq \begin{bmatrix}\mathbf{\Sigma}^T & \mathbf{V}^T\end{bmatrix}^T$, where the rows of $\mathbf{V} \in \mathbb{R}^{(n-r^{\star})\times n}$ represent an orthogonal basis for the null space of $\mathbf{\Sigma}$. Using (2.57), we then conclude that

$$\mathbf{TA} = \begin{bmatrix}\mathbf{\Sigma A}\\ \mathbf{VA}\end{bmatrix} = \begin{bmatrix}\mathbf{A}_D\mathbf{\Sigma}\\ \mathbf{VA}\end{bmatrix} = \begin{bmatrix}\begin{bmatrix}\mathbf{A}_D & \mathbf{0}\end{bmatrix}\mathbf{T}\\ \mathbf{VAT}^{-1}\mathbf{T}\end{bmatrix}. \tag{2.58}$$

By partitioning the $(n-r^{\star})\times n$ matrix $\mathbf{VAT}^{-1}$ as $\mathbf{VAT}^{-1} = \begin{bmatrix}\mathbf{A}_E & \mathbf{A}_F\end{bmatrix}$, where $\mathbf{A}_E \in \mathbb{R}^{(n-r^{\star})\times r^{\star}}$ and $\mathbf{A}_F \in \mathbb{R}^{(n-r^{\star})\times(n-r^{\star})}$, we further obtain

$$\mathbf{TA} = \begin{bmatrix}\mathbf{A}_D & \mathbf{0}\\ \mathbf{A}_E & \mathbf{A}_F\end{bmatrix}\mathbf{T}. \tag{2.59}$$

Next, note that $\mathcal{R}(\mathbf{C}^{\star}) \subseteq \mathcal{R}(\mathbf{\Sigma})$. Hence, there exists a matrix $\mathbf{C}_D \in \mathbb{R}^{p\times r^{\star}}$ such that

$$\mathbf{C}^{\star} = \mathbf{C}_D\mathbf{\Sigma}. \tag{2.60}$$

Noting that $\mathbf{\Sigma T}^{-1} = \begin{bmatrix}\mathbf{I} & \mathbf{0}\end{bmatrix}$, and using (2.60), we see that

$$\mathbf{C}^{\star}\mathbf{T}^{-1} = \begin{bmatrix}\mathbf{C}_D & \mathbf{0}\end{bmatrix}. \tag{2.61}$$

Defining $\bar{\mathbf{A}} \triangleq \mathbf{TAT}^{-1}$, $\bar{\mathbf{C}} \triangleq \mathbf{C}^{\star}\mathbf{T}^{-1}$, and using (2.59) and (2.61), we obtain (2.56). It remains to show that (2.54) implies detectability of the pair $(\mathbf{A}_D, \mathbf{C}_D)$. To this end, note that the unique solutions to (2.57) and (2.60) are given by $\mathbf{A}_D = \mathbf{\Sigma A \Sigma}^{\dagger}$ and $\mathbf{C}_D = \mathbf{C}^{\star}\mathbf{\Sigma}^{\dagger}$, respectively. Based on the PBH test, the pair $(\mathbf{A}_D, \mathbf{C}_D)$ is detectable if and only if

$$\operatorname{rank}\begin{bmatrix}s\mathbf{I} - \mathbf{\Sigma A \Sigma}^{\dagger}\\ \mathbf{C}^{\star}\mathbf{\Sigma}^{\dagger}\end{bmatrix} = r^{\star}, \quad \forall s \in \mathbb{C}, \quad |s| \geq 1. \tag{2.62}$$

Since $\begin{bmatrix} \mathbf{\Sigma}^{\dagger} & \mathbf{I}-\mathbf{\Sigma}^{\dagger}\mathbf{\Sigma} \end{bmatrix}$ is full row-rank, it follows that

$$\begin{aligned} \text{rank} \begin{bmatrix} s\mathbf{\Sigma}-\mathbf{\Sigma}\mathbf{A} \\ \mathbf{C}^{\star} \end{bmatrix} &= \text{rank} \begin{bmatrix} s\mathbf{\Sigma}-\mathbf{\Sigma}\mathbf{A} \\ \mathbf{C}^{\star} \end{bmatrix} \begin{bmatrix} \mathbf{\Sigma}^{\dagger} & \mathbf{I}-\mathbf{\Sigma}^{\dagger}\mathbf{\Sigma} \end{bmatrix} \\ &= \text{rank} \begin{bmatrix} s\mathbf{I}-\mathbf{\Sigma}\mathbf{A}\mathbf{\Sigma}^{\dagger} & \mathbf{0} \\ \mathbf{C}^{\star}\mathbf{\Sigma}^{\dagger} & \mathbf{0} \end{bmatrix}. \end{aligned} \tag{2.63}$$

The last equality follows by noting that the matrix $(\mathbf{I}-\mathbf{\Sigma}^{\dagger}\mathbf{\Sigma})$ projects onto the null space of $\mathbf{\Sigma}$, and that $\mathcal{R}(\mathbf{\Sigma}\mathbf{A})$, $\mathcal{R}(\mathbf{C}^{\star})$ are both contained in $\mathcal{R}(\mathbf{\Sigma})$. Finally, combining (2.54), (2.62) and (2.63) leads to the desired result. ■

Remark 2.9.5 *Note that Lemma 2.9.4 does not describe a standard detectable decomposition of the pair $(\mathbf{A},\mathbf{C}^{\star})$. In fact, the dimension of the square matrix $\mathbf{A}_D$, namely $r^{\star}$, is in general smaller than the dimensions of the detectable subspaces of the pairs $(\mathbf{A},\mathbf{C}^{\star})$, $(\mathbf{A},\mathbf{C}_{\mathcal{S}^{\star}})$ and $(\mathbf{A},\mathbf{C})$.*

Based on (2.56), we obtain the following dynamics:

$$\boldsymbol{\phi}[k+1]=\mathbf{A}_D\boldsymbol{\phi}[k], \bar{\mathbf{y}}[k]=\mathbf{C}_D\boldsymbol{\phi}[k], \tag{2.64}$$

where $\boldsymbol{\phi}[k]=\begin{bmatrix} \mathbf{I}_{r^{\star}} & \mathbf{0} \end{bmatrix}\mathbf{T}\mathbf{x}[k]=\mathbf{\Sigma}\mathbf{x}[k]$, $\bar{\mathbf{y}}[k]$ represents measurements of the sensor nodes in $\mathcal{S}^{\star}$ corresponding to the matrix $\mathbf{C}^{\star}$, i.e., $\bar{\mathbf{y}}[k]=\mathbf{C}^{\star}\mathbf{x}[k]$, and $\mathbf{A}_D$, $\mathbf{C}_D$, $\mathbf{\Sigma}=\begin{bmatrix} \mathbf{L}^T & \tilde{\mathbf{C}}^{\star T} \end{bmatrix}^T$ are as described by Lemma 2.9.4. In particular, note that the first r states of the vector $\boldsymbol{\phi}[k]$ represent the functionals of interest, namely $\boldsymbol{\psi}[k]$. Effectively, we have converted the distributed functional estimation problem to the problem of designing a full-order distributed state observer for the state $\boldsymbol{\phi}[k]$ described by (2.64). Thus, if the pair $(\mathbf{A}_D,\mathbf{C}_D)$ is detectable, one can leverage the distributed observer design approach developed earlier in the chapter to enable each node to recover $\boldsymbol{\phi}[k]$, and hence $\boldsymbol{\psi}[k]$. The above discussion immediately leads to the main result of this section.

Theorem 2.9.6 *Given a tuple $(\mathbf{A},\mathbf{C},\mathbf{L})$ described by equations (2.1), (2.2), and (2.47), and a strongly-connected communication graph $\mathcal{G}$, let the feasible leader set*

$\mathcal{F}$ described by Definition 2.9.1 be non-empty. Then, the proposed distributed functional observer described by the update equations (2.10), (2.11), and (2.13) solves Problem 2.

Proof Since $\mathcal{F}$ is non-empty, there exists a functional leader set $\mathcal{S}^\star$. Based on the property of $\mathcal{S}^\star$ described by Lemma 2.9.4, the pair $(\mathbf{A}_D, \mathbf{C}_D)$ governing the dynamics of $\boldsymbol{\phi}[k]$ (see (2.64)) is detectable. Since $\mathcal{G}$ is strongly-connected, one can use the distributed observer design approach outlined in Section 2.5.2 to enable each node in $\mathcal{V}$ to asymptotically estimate $\boldsymbol{\phi}[k]$. Noting that the desired functionals $\boldsymbol{\psi}[k]$ satisfy $\boldsymbol{\psi}[k] = \begin{bmatrix}\mathbf{I}_r & \mathbf{0}\end{bmatrix} \boldsymbol{\phi}[k]$ completes the proof. ■

Remark 2.9.7 *Note that the order of the proposed distributed functional observer is $r^\star$ where $r \leq r^\star \leq d$ in general, with $r = rank\,(\mathbf{L})$ and d equal to the dimension of the detectable subspace of the pair $(\mathbf{A}, \mathbf{C})$ given by (2.1) and (2.2).*[21] *The fact that $r^\star \geq r$ follows from the discussion in Section 2.9.2. For the special case when $\mathcal{R}(\mathbf{C}^\star) \subseteq \mathcal{R}(\mathbf{L})$, we have $r^\star = r$. Further, when the tuple $(\mathbf{A}, \mathbf{C}, \mathbf{L})$ is 'functionally observable' [66], i.e., when the functionals of interest are linear combinations of only the observable states of $(\mathbf{A}, \mathbf{C})$, it is easily seen that the order of the proposed observer is no greater than the dimension of the observable subspace of $(\mathbf{A}, \mathbf{C})$.*

Remark 2.9.8 *Note that when $\mathbf{L} = \mathbf{I}_n$, the rank condition (2.53) is trivially satisfied whereas (2.54) boils down to the existence of a set of nodes $\mathcal{S} \in \mathcal{V}$ such that the pair $(\mathbf{A}, \mathbf{C}_{\mathcal{S}})$ is detectable. For a strongly connected graph $\mathcal{G}$, it was shown earlier in the chapter (and in [1, 30]) that the necessary and sufficient condition for distributed state estimation is the detectability of the pair $(\mathbf{A}, \mathbf{C})$. Thus, it is apparent that for a strongly connected graph $\mathcal{G}$, the sufficient condition presented in this section for the construction of a distributed functional observer, namely that the feasible leader set $\mathcal{F}$ is non-empty, is in fact a generalization of the aforementioned necessary and sufficient condition for distributed state estimation.*

[21] Note that if $\mathcal{F}$ is non-empty, then a centralized functional observer of order r can always be constructed.

2.10 Chapter Summary

In this chapter, we first considered the problem of distributed state estimation of an LTI system by a network of nodes. We introduced a new class of distributed observers for the most general class of LTI systems, time-invariant directed communication graphs, and linear sensor measurement structures. This was achieved by extending the Kalman observable canonical decomposition to a setting with multiple sensors, i.e., by introducing the notion of a multi-sensor observable canonical decomposition.

We also demonstrated that for certain subclasses of system dynamics and networks, one can design a distributed observer via a simpler estimation scheme which enjoys the benefit of a fully distributed design phase. We then discussed how our proposed framework can be extended to account for communication failures. The main underlying theme of our work is built upon the following intuition: portions of the state space can be reconstructed by a node using its own local measurements, and hence it needs to run consensus for only the portion of the state space that is not locally detectable. In the latter half of the chapter, we generalized our results to address the problem of distributed functional estimation.

2.11 Omitted Proofs

2.11.1 Proof of Proposition 2.5.1

Proof We outline the sequence of transformations that need to be carried out.

Step 1 : *Transformation at Sensor 1*

Consider the coordinate transformation $\mathbf{x}[k] = \mathbf{T}_1\mathbf{z}_1[k]$. Here, $\mathbf{T}_1$ is the non-singular matrix that performs an observable canonical decomposition of the pair $(\mathbf{A}, \mathbf{C}_1)$, yielding

$$\begin{aligned} \underbrace{\begin{bmatrix} \mathbf{z}^{(1)}[k+1] \\ \mathbf{z}^{(1)}_{\mathcal{U}}[k+1] \end{bmatrix}}_{\mathbf{z}_1[k+1]} &= \underbrace{\begin{bmatrix} \mathbf{A}_{11} & \mathbf{0} \\ \mathbf{A}_{1X} & \mathbf{A}_{1\mathcal{U}} \end{bmatrix}}_{\bar{\mathbf{A}}_1 = \mathbf{T}_1^{-1}\mathbf{A}\mathbf{T}_1} \underbrace{\begin{bmatrix} \mathbf{z}^{(1)}[k] \\ \mathbf{z}^{(1)}_{\mathcal{U}}[k] \end{bmatrix}}_{\mathbf{z}_1[k]}, \\ \mathbf{y}_1[k] &= \underbrace{\begin{bmatrix} \mathbf{C}_{11} & \mathbf{0} \end{bmatrix}}_{\bar{\mathbf{C}}_1 = \mathbf{C}_1\mathbf{T}_1} \mathbf{z}_1[k]. \end{aligned} \tag{2.65}$$

Let $\mathbf{z}^{(1)}[k] \in \mathbb{R}^{o_1}$. From (2.65), we obtain

$$\begin{aligned} \mathbf{z}^{(1)}[k+1] &= \mathbf{A}_{11}\mathbf{z}^{(1)}[k], \\ \mathbf{y}_1[k] &= \mathbf{C}_{11}\mathbf{z}^{(1)}[k]. \end{aligned} \tag{2.66}$$

Step 2 : *Transformation at Sensor 2*

We know the following:

$$\begin{aligned} \mathbf{y}_2[k] &= \mathbf{C}_2\mathbf{x}[k] = \mathbf{C}_2\mathbf{T}_1\mathbf{z}_1[k] \\ &\triangleq \begin{bmatrix} \mathbf{C}_{21} & \mathbf{C}_{21\mathcal{U}} \end{bmatrix} \begin{bmatrix} \mathbf{z}^{(1)}[k] \\ \mathbf{z}^{(1)}_{\mathcal{U}}[k] \end{bmatrix}. \end{aligned} \tag{2.67}$$

Let $\bar{\mathbf{T}}_2$ be a non-singular transformation matrix that performs an observable canonical decomposition of the pair $(\mathbf{A}_{1\mathcal{U}}, \mathbf{C}_{21\mathcal{U}})$. We now wish to identify the portion of the unobservable subspace of sensor 1 that is observable with respect to sensor 2. With this objective in mind, consider the coordinate transformation $\mathbf{z}_1[k] = \mathbf{T}_2\mathbf{z}_2[k]$, where the non-singular transformation matrix $\mathbf{T}_2$ is defined as

$$\mathbf{T}_2 = \begin{bmatrix} \mathbf{I}_{o_1} & \mathbf{0} \\ \mathbf{0} & \bar{\mathbf{T}}_2 \end{bmatrix}. \tag{2.68}$$

This yields the following dynamics:

$$\underbrace{\begin{bmatrix} \mathbf{z}^{(1)}[k+1] \\ \mathbf{z}^{(2)}[k+1] \\ \mathbf{z}^{(2)}_{\mathcal{U}}[k+1] \end{bmatrix}}_{\mathbf{z}_2[k+1]} = \underbrace{\left[\begin{array}{c|cc} \mathbf{A}_{11} & \multicolumn{2}{c}{\mathbf{0}} \\ \hline & \mathbf{A}_{22} & \mathbf{0} \\ \bar{\mathbf{T}}_2^{-1}\mathbf{A}_{1X} & & \\ & \star & \mathbf{A}_{2\mathcal{U}} \end{array}\right]}_{\bar{\mathbf{A}}_2 = \mathbf{T}_2^{-1}\bar{\mathbf{A}}_1\mathbf{T}_2} \underbrace{\begin{bmatrix} \mathbf{z}^{(1)}[k] \\ \mathbf{z}^{(2)}[k] \\ \mathbf{z}^{(2)}_{\mathcal{U}}[k] \end{bmatrix}}_{\mathbf{z}_2[k]},$$
$$\mathbf{y}_2[k] = \underbrace{\left[\begin{array}{c|cc} \mathbf{C}_{21} & \mathbf{C}_{22} & \mathbf{0} \end{array}\right]}_{\bar{\mathbf{C}}_2 = \mathbf{C}_2\mathbf{T}_1\mathbf{T}_2} \mathbf{z}_2[k], \tag{2.69}$$

where

$$\bar{\mathbf{T}}_2^{-1}\mathbf{A}_{1\mathcal{U}}\bar{\mathbf{T}}_2 = \begin{bmatrix} \mathbf{A}_{22} & \mathbf{0} \\ \star & \mathbf{A}_{2\mathcal{U}} \end{bmatrix},$$
$$\mathbf{C}_{21\mathcal{U}}\bar{\mathbf{T}}_2 = \begin{bmatrix} \mathbf{C}_{22} & \mathbf{0} \end{bmatrix}. \tag{2.70}$$

Let $\mathbf{z}^{(2)}[k] \in \mathbb{R}^{o_2}$. Let $\mathbf{A}_{21}$ be the matrix formed by the first o_2 rows of $\bar{\mathbf{T}}_2^{-1}\mathbf{A}_{1X}$. From (2.69), we have

$$\mathbf{z}^{(2)}[k+1] = \mathbf{A}_{22}\mathbf{z}^{(2)}[k] + \mathbf{A}_{21}\mathbf{z}^{(1)}[k],$$
$$\mathbf{y}_2[k] = \mathbf{C}_{22}\mathbf{z}^{(2)}[k] + \mathbf{C}_{21}\mathbf{z}^{(1)}[k]. \tag{2.71}$$

Following the same design procedure, we continue the sequence of transformations, one for each sensor, until we reach the N-th sensor.

Step N : *Transformation at Sensor* N

Let $\bar{\mathbf{T}}_N$ be a transformation matrix that performs an observable canonical decomposition of the pair $(\mathbf{A}_{(N-1)\mathcal{U}}, \mathbf{C}_{N(N-1)\mathcal{U}})$. Next, consider the coordinate transformation $\mathbf{z}_{N-1}[k] = \mathbf{T}_N\mathbf{z}_N[k]$, where the transformation matrix $\mathbf{T}_N$ is defined as follows:

$$\mathbf{T}_N = \left[\begin{array}{c|c|cc} \mathbf{I}_{o_1} & \multicolumn{3}{c}{\mathbf{0}} \\ \hline & \mathbf{I}_{o_2} & \multicolumn{2}{c}{\mathbf{0}} \\ \cline{2-4} & & \ddots & \vdots \\ \mathbf{0} & \mathbf{0} & \mathbf{I}_{o_{(N-1)}} & \mathbf{0} \\ & & \mathbf{0} & \bar{\mathbf{T}}_N \end{array}\right]. \tag{2.72}$$

Using this transformation matrix, it is easy to identify that the resulting dynamics are governed by the following equations:

$$\begin{aligned} \mathbf{z}_N[k+1] &= \bar{\mathbf{A}}_N \mathbf{z}_N[k], \\ \mathbf{y}_N[k] &= \bar{\mathbf{C}}_N \mathbf{z}_N[k], \end{aligned} \tag{2.73}$$

where $\bar{\mathbf{A}}_N$ equals $\bar{\mathbf{A}}$ in equation (2.5) and $\bar{\mathbf{C}}_N$ attains the form:

$$\bar{\mathbf{C}}_N = \left[\begin{array}{cccc|c} \mathbf{C}_{N1} & \mathbf{C}_{N2} & \cdots \mathbf{C}_{N(N-1)} & \mathbf{C}_{NN} & \mathbf{0} \end{array} \right]. \tag{2.74}$$

Thus, by defining $\mathcal{T} \triangleq \prod_{i=1}^{n} \mathbf{T}_i$, we obtain the desired result. ■

2.11.2 Proof of Theorem 2.7.2

Proof Following the proof technique of Theorem 2.5.6, we induct on the sub-state number and use the same notation as in the former proof. Accordingly, note that the dynamics of the composite estimation error vector for the first sub-state, namely $\bar{\mathbf{E}}^{(1)}[k]$, is governed by the following switched linear system model: $\bar{\mathbf{E}}^{(1)}[k+1] = \mathbf{M}_1[k]\bar{\mathbf{E}}^{(1)}[k]$. Note that the entries of $\bar{\mathbf{E}}^{(1)}[k]$ match a topological ordering consistent with a spanning DAG rooted at node 1 in the baseline graph $\mathcal{G}$. Here, $\mathbf{M}_1[k]$ is a time-varying matrix induced by the class of switching signals Ω and is of the structure given by (2.22). Since Ω satisfies Assumption 2.7.1, each non-source node $i \in \mathcal{V} \setminus \{1\}$ is guaranteed to receive information from at least one of its parents in $\mathcal{P}_i^{(1)}$ in at least one switching mode over every time interval of the form $[kT, (k+1)T)$, where $k \in \mathbb{N}$. Based on our estimation scheme, for that corresponding switching mode, the block diagonal entry corresponding to node i in the matrix $\mathbf{M}_1[k]$ will be zero. With this observation in mind, consider the following dynamics: $\bar{\mathbf{E}}^{(1)}[(k+1)T] = \bar{\mathbf{M}}_1(k)\bar{\mathbf{E}}^{(1)}[kT]$, where $\bar{\mathbf{M}}_1(k) = \mathbf{M}_1[(k+1)T-1] \cdots \mathbf{M}_1[kT+1]\mathbf{M}_1[kT]$. From our prior discussion, it easily follows that $\bar{\mathbf{M}}_1(k)$ is a lower block triangular matrix with zeroes on the block-diagonal corresponding to the non-source nodes in $\mathcal{V} \setminus \{1\}$ and the entry $(\mathbf{A_{11}} - \mathbf{L_1 C_{11}})^T$ corresponding to node 1. As the pair $(\mathbf{A}_{11}, \mathbf{C}_{11})$ is observable by construction, it follows using standard arguments that $\bar{\mathbf{M}}_1(k)$ is always a Schur stable matrix. Since

$\bar{\mathbf{M}}_1(k)$ belongs to a finite set of matrices (owing to a finite number of switching modes), we can directly use [68, Proposition 2.9] to establish that $\lim_{k\to\infty} \mathbf{E}^{(1)}[kT] = \mathbf{0}$ and hence $\lim_{k\to\infty} \mathbf{E}^{(1)}[k] = \mathbf{0}$. Next, suppose that $\mathbf{E}^{(j)}[k]$ converges to zero asymptotically $\forall j \in \{1, \cdots, p-1\}$, where $1 \leq p-1 \leq N-1$. The composite estimation error dynamics for sub-state p over an interval of length T is given by

$$\bar{\mathbf{E}}^{(p)}[(k+1)T] = \bar{\mathbf{M}}_p(k)\bar{\mathbf{E}}^{(p)}[kT] + \bar{\mathbf{F}}_p(k)\bar{\mathbf{v}}^{(p)}, \tag{2.75}$$

where $\bar{\mathbf{M}}_p(k)$ is defined in the same way as $\bar{\mathbf{M}}_1(k)$, and

$$\bar{\mathbf{v}}^{(p)} = \begin{bmatrix} \mathbf{v}^{(p)}[kT] \\ \vdots \\ \mathbf{v}^{(p)}[(k+1)T-1] \end{bmatrix}, \mathbf{v}^{(p)}[k] = \sum_{l=1}^{p-1} \mathbf{H}_{pl}\bar{\mathbf{E}}^{(pl)}[k],$$

$$\bar{\mathbf{F}}_p(k) = \Big[(\mathbf{M}_p[(k+1)T-1]\cdots\mathbf{M}_p[kT+1]) \quad \cdots \quad \mathbf{M}_p[(k+1)T-1] \quad \mathbf{I}_{No_p}\Big].$$

It follows from our induction hypothesis that $\lim_{k\to\infty} \mathbf{v}^{(p)}[k] = \mathbf{0}$. Since $\bar{\mathbf{M}}_p(k)$ is Schur stable for the class of switching signals satisfying Assumption 2.7.1 (in the same way as $\bar{\mathbf{M}}_1(k)$ is Schur stable), it follows from ISS and [68, Proposition 2.9] that $\lim_{k\to\infty} \mathbf{E}^{(p)}[k] = \mathbf{0}$. Since the update rule (2.13) for the unobservable component of the state is unaffected by changes in the network structure, the rest of the proof proceeds similarly as the proof of Theorem 2.5.6. ■

3. BYZANTINE-RESILIENT DISTRIBUTED OBSERVERS FOR LTI SYSTEMS

In this chapter, we study the problem of collaboratively estimating the state of an LTI system when certain nodes are compromised by adversaries. Specifically, we consider a Byzantine adversary model, where a compromised node possesses complete knowledge of the system dynamics and the network, and can deviate arbitrarily from the rules of any prescribed algorithm. We first characterize certain fundamental limitations of any distributed state estimation algorithm in terms of the measurement and communication structure of the nodes. We then develop an attack-resilient, provably correct state estimation algorithm that admits a fully distributed implementation. To characterize feasible network topologies that guarantee applicability of our proposed technique, we introduce a notion of 'strong-robustness' that captures both measurement and communication redundancy. Finally, by drawing connections to bootstrap percolation theory, we argue that given an LTI system and an associated sensor network, the 'strong-robustness' property can be checked in polynomial time.

3.1 Introduction

The control of large-scale complex networked systems such as power grids, transportation networks, and multi-agent robotic systems requires precise estimation of the state of the underlying dynamical process. Typically, in these applications, sensors (nodes) collecting information about the process are scattered over a geographical region. As the diameters of such networks increase, routing information from all the sensors to a central computational resource induces large delays and creates communication bottlenecks. To bypass these difficulties, it thus becomes important to consider distributed algorithms where individual sensors communicate only with sensors within a given distance. However, the potential merits [24] of such a distributed approach are matched by various challenges. In particular, a key challenge is to design networks and distributed algorithms that guarantee reliable operation of the system in the face of faults or sophisticated adversarial attacks on certain sensors. The existing distributed state estimation approaches that we surveyed in Chapter 2 do not account for such scenarios. This motivates the content of the present chapter.

3.1.1 Related Work

Over the last decade, a significant amount of research has focused on security in networked control systems. In particular, for noiseless dynamical systems, it has been established that zero-dynamics play a key role in characterizing the stealth of an attack [69,70]. For networked control systems affected by noise, the authors in [71] recently introduced an information-theoretic metric that quantifies the detectability of an attack. A unifying feature of [69, 71], and the ones in [72–75], is that they involve systems where all the sensor measurements are available at a single location. In the sequel, we shall refer to such systems as centralized control systems. Our problem formulation and subsequent analysis differs from the above literature by constraining each sensor to exchange information with only its neighbors in the communication graph. Some recent related work on resilient distributed parameter estimation and resilient decentralized hypothesis testing are reported in [76] and [77], respectively. The authors in [78] consider the problem of joint attack detection and state estimation. However, the attack model, the system model, and the assumptions on the communication graph in [78] differ considerably from the ones considered in this chapter.

While the study of security in centralized control systems is now mature, there lacks a comprehensive theoretical understanding of analogous questions in a distributed setting. Preliminary attempts to counter adversarial behavior in a distributed state estimation context are reported in [79], [80]. However, unlike our results, these papers neither provide any theoretical guarantees of success, nor allude to graph-theoretic conditions that are necessary for their respective algorithms to work. Recently, in [81], the authors employ an H_∞ based approach for detecting biasing attacks in distributed estimation networks. Our present work deviates from [81] in several aspects, namely (i) while the analysis in [81] is limited to a certain class of attack inputs, our attack model allows compromised nodes to behave *arbitrarily*, i.e., no restrictions are placed on the inputs that can be injected by an adversary, (ii) unlike [81], we develop a filtering algorithm that allows each uncompromised node to asymptotically recover the state of the plant *without* explicitly detecting the nodes under attack, and (iii) the existence of the attack detection filter proposed in [81] relies on solving an LMI; however, the authors neither provide graph-theoretic insights regarding the solvability of such an LMI nor discuss whether the LMI can be solved in a distributed manner. In contrast, we detail graph-theoretic conditions that allow each step of our approach to have a resilient, distributed implementation. Summing up, this chapter attempts to bridge the gap between centralized and distributed resilient state estimation. Our main contributions in this context are discussed below.

3.1.2 Summary of Contributions

Our contributions are threefold. First, in Section 3.3, we characterize certain necessary conditions that need to be satisfied by the sensor measurements and the communication graph for the distributed state estimation problem to be solvable in the presence of arbitrary adversarial behavior. Our results hold for *any* algorithm and hence identify fundamental limitations that are of both theoretical and practical importance in the design of attack-resilient robust networks. We also argue that our impossibility results in the distributed setting generalize those existing for centralized control systems subject to sensor attacks [82, 83].

For the problem under consideration, it is imperative to understand which (potentially adversarial) neighbors a given node should listen to, and subsequently, how it should process the information received from neighbors it chooses to listen to. Consequently, our second contribution is to develop a distributed filtering algorithm in Section 3.4 that enables each uncompromised node to recover the entire state dynamics, provided certain graph conditions are met. A thorough analysis of the proposed filtering scheme is then presented in Section 3.5.

As our third contribution, in Section 3.7, we introduce a topological property called 'strong-robustness' to characterize feasible systems and networks that guarantee applicability of our approach. By drawing connections to bootstrap percolation theory, we show that the 'strong-robustness' property can be checked in polynomial time (in the size of the system and the network).

The results in this chapter were published as [84] and [85].

Throughout this chapter, the term 'resilient' will be used in the same context as that used traditionally in the computer science literature to deal with worst-case adversarial attack models [86].

3.2 System and Attack Model

System Model: We consider the LTI system model (2.1) and the observation model (2.2) studied earlier in Chapter 2. Recall that each node is tasked with estimating the entire system state $\mathbf{x}[k]$ based on information received from its neighbors and its local measurements (if any). As such, we assume that the pair $(\mathbf{A}, \mathbf{C})$ is detectable (this is a necessary condition for solving the distributed state estimation problem even in the absence of adversaries); however, we do not assume that the pair $(\mathbf{A}, \mathbf{C}_i)$ is detectable for any $i \in \mathcal{V}$. Two immediate challenges (as identified in Chapter 2) are as follows: (i) As the pair $(\mathbf{A}, \mathbf{C}_i)$ may not be detectable for some (or all) $i \in \{1, \cdots, N\}$, information exchange is necessary; and (ii) information exchange is restricted by the underlying communication graph $\mathcal{G}$. In addition to the above challenges, in this chapter, we allow for the possibility that certain nodes in the network are compromised by an adversary, and *do not* follow their prescribed state estimate update rule. We will use the following adversary model in this chapter.

Adversary Model: We consider a subset $\mathcal{A} \subset \mathcal{V}$ of the nodes in the network to be adversarial. We assume that the adversarial nodes are completely aware of the network topology, the system dynamics, and the algorithm employed by the non-adversarial nodes. Such an assumption of omniscient adversarial behavior is standard in the literature on resilient distributed algorithms [87–95], and allows us to provide guarantees against "worst-case" adversarial behavior. In terms of capabilities, an adversarial node can leverage the aforementioned information to arbitrarily deviate from the rules of any prescribed algorithm, while colluding with other adversaries in the process. Furthermore, following the Byzantine fault model [86], adversaries are allowed to send differing state estimates to different neighbors at the same instant of time. To characterize the threat model in terms of the number of adversaries in the network, we will use the following definitions from [91], [92].

Definition 3.2.1 *(f**-total set**) A set $\mathcal{C} \subset \mathcal{V}$ is f-total if it contains at most f nodes in the network, i.e., $|\mathcal{C}| \leq f$.*

Definition 3.2.2 *(f**-local set**) A set $\mathcal{C} \subset \mathcal{V}$ is f-local if it contains at most f nodes in the neighborhood of the other nodes, i.e., $|\mathcal{N}_i \cap \mathcal{C}| \leq f$, $\forall i \in \mathcal{V} \setminus \mathcal{C}$.*

Definition 3.2.3 *(f**-local and** f**-total adversarial models**) A set $\mathcal{A}$ of adversarial nodes is f-locally bounded (resp., f-totally bounded) if $\mathcal{A}$ is an f-local (resp., f-total) set.*

In the literature dealing with distributed fault-tolerant algorithms, it is a common assumption to consider an f-total adversarial model. However, to allow for a large number of adversaries in large scale networks, we will allow the adversarial set to be f-local. Summarily, the adversary model considered throughout this chapter will be referred to as an f-locally bounded Byzantine adversary model. The non-adversarial nodes will be referred to as regular nodes and be represented by the set $\mathcal{R} = \mathcal{V} \setminus \mathcal{A}$.

Note that the actual number and identities of the adversarial nodes are not known to the regular nodes. As is standard, any reliable system is designed to provide a desired level of resilience against a maximum number of component failures or attacks. We share the same philosophy. Specifically, we assume that each node in the network is programmed to tolerate upto a maximum of f adversaries in the entire network (in an f-total model) or in its own neighborhood (in an f-local model). Such an assumption is typical in the design of distributed protocols (for varied applications, such as consensus [87,88,96], optimization [89,90], reference-tracking [93], formation-control [94], multi-agent rendezvous [95], and broadcasting [91,92]) that are resilient to worst-case Byzantine models like the one considered in this chapter.[1]

Throughout this chapter, we shall only consider causal (i.e., nodes act only on past and present information), synchronous (i.e., all nodes share a common clock w.r.t. their iterates), and deterministic algorithms (i.e., given the same input, such algorithms generate the same output); note however that the notions of causal and deterministic behavior apply only to the regular nodes. We shall also assume that all quantities being updated iteratively by the regular nodes are initialized identically in each execution. With $\hat{\mathbf{x}}_i[k]$ representing the estimate of $\mathbf{x}[k]$ maintained by node i, the problem studied in this chapter can be formally stated as follows.

Problem 3 ***(Resilient Distributed State Estimation)*** *Given an LTI system (2.1), a linear measurement model (2.2), and a time-invariant directed communication graph $\mathcal{G}$, design a set of state estimate update and information exchange rules such that $\lim_{k\to\infty} \|\hat{\mathbf{x}}_i[k] - \mathbf{x}[k]\| = 0$, $\forall i \in \mathcal{R}$, regardless of the actions of any f-locally bounded set of Byzantine adversaries.*

The interplay between the measurement structure of the nodes and the underlying communication graph results in certain conditions being necessary for solving Problem 3, irrespective of the choice of algorithm. We provide such conditions in the following section.

[1] Some recent papers that look at weaker adversarial models than those considered by us are [97–99].

3.3 Fundamental Limitations of any Distributed State Estimation Algorithm

Intuitively, the network must possess a certain degree of measurement redundancy as well as redundancy in its communication structure so as to counteract the effects of adversarial behavior. More specifically, the measurements of the regular nodes must ensure collective detectability of the state, and the network structure should prevent the malicious nodes from acting as bottlenecks between correctly functioning nodes. To identify necessary conditions for resilient distributed state estimation that capture the above notions of redundancy, we first introduce some terminology.

Definition 3.3.1 *(**Critical Set**) A set of nodes $\mathcal{F} \subset \mathcal{V}$ is said to be a critical set if the pair $(\mathbf{A}, \mathbf{C}_{\mathcal{V}\setminus\mathcal{F}})$ is not detectable.*

Note that detectability of $(\mathbf{A}, \mathbf{C})$ implies that a critical set must necessarily be non-empty.

Definition 3.3.2 *(**Minimal Critical Set**) A set $\mathcal{F} \subset \mathcal{V}$ is said to be a minimal critical set if $\mathcal{F}$ is a critical set and no subset of $\mathcal{F}$ is a critical set.*

Let $\mathcal{M} = \{\mathcal{F}_1, \cdots, \mathcal{F}_{|\mathcal{M}|}\}$ denote the set of all minimal critical sets. With each set $\mathcal{F}_i \in \mathcal{M}$, we associate a virtual node s_i as follows. Directed edges are added from s_i to each node in $\mathcal{F}_i$ and the resulting network is denoted by $\mathcal{G}_i' = (\mathcal{V} \cup s_i, \mathcal{E} \cup \mathcal{E}_i)$, where $\mathcal{E}_i$ represents the set of edges from s_i to $\mathcal{F}_i$.

Definition 3.3.3 *(f-**local pair and** f-**total pair cuts w.r.t.** s_i) Consider a minimal critical set $\mathcal{F}_i \in \mathcal{M}$. A set $\mathcal{H} \subset \mathcal{V}$ is called a cut w.r.t. s_i if removal of $\mathcal{H}$ from $\mathcal{G}_i'$ results in an induced subgraph of $\mathcal{G}_i'$ whose node set can be partitioned into two non-empty sets $\mathcal{X}$ and $\mathcal{Y}$ with $s_i \in \mathcal{X}$, and no directed paths from $\mathcal{X}$ to $\mathcal{Y}$ in the induced subgraph. A cut $\mathcal{H}$ w.r.t. s_i is called an f-local pair cut (resp., f-total pair cut) w.r.t. s_i if it can be partitioned as $\mathcal{H} = \mathcal{H}_1 \cup \mathcal{H}_2$ such that both $\mathcal{H}_1$ and $\mathcal{H}_2$ are f-local (resp., f-total) in $\mathcal{G}$.*

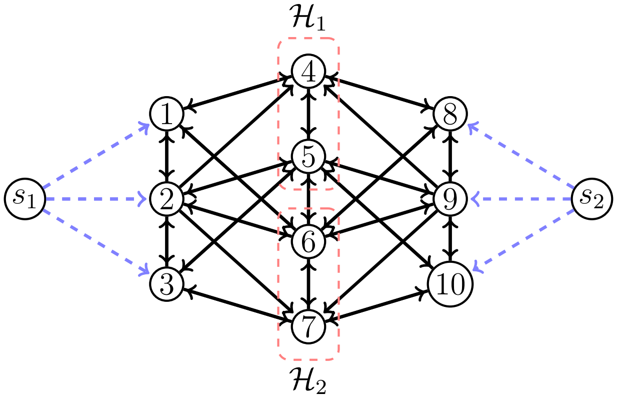


Fig. 3.1. A 2-dimensional LTI system with two distinct, real, unstable eigenvalues (modes) λ_1, λ_2 is monitored by a network $\mathcal{G}$ of 10 nodes as shown above. Nodes 1-3 can detect λ_1, while nodes 8-10 can detect λ_2. Thus, the two minimal critical sets associated with the above system and network are $\mathcal{F}_1 = \{1, 2, 3\}$ and $\mathcal{F}_2 = \{8, 9, 10\}$. An example of a set that is critical, but not minimal, is $\{1, 2, 3, 8\}$. The virtual source nodes associated with $\mathcal{F}_1$ and $\mathcal{F}_2$ are s_1 and s_2, respectively. There are no 1-total pair cuts w.r.t. s_1 or s_2. The set $\mathcal{H} = \{4, 5, 6, 7\}$ is a 1-local pair cut w.r.t. both s_1 and s_2 since $\mathcal{H}$ can be partitioned into $\mathcal{H}_1 = \{4, 5\}$ and $\mathcal{H}_2 = \{6, 7\}$, each of which are 1-local sets. Since $\mathcal{H}_1$ and $\mathcal{H}_2$ are each 2-total sets, $\mathcal{H}$ is also a 2-total pair cut w.r.t. both s_1 and s_2.

For an illustration of the above definitions, see Figure 3.1. The following result identifies a fundamental limitation for f-local adversarial models.

Theorem 3.3.1 *Suppose there exists an f-local pair cut w.r.t. s_i in $\mathcal{G}_i'$ for some minimal critical set $\mathcal{F}_i \in \mathcal{M}$. Then, it is impossible for any causal, synchronous and deterministic algorithm to solve Problem 3.*

Proof Suppose there exists an f-local pair cut $\mathcal{H} = \mathcal{H}_1 \cup \mathcal{H}_2$ w.r.t. s_i for some minimal critical set $\mathcal{F}_i \in \mathcal{M}$. For the sake of contradiction, suppose there exists a causal, synchronous and deterministic algorithm $\mathcal{T}$ that solves Problem 3 for the given network $\mathcal{G}$. From the definition of $\mathcal{H}$, we see that $\mathcal{Y}$ contains no elements of $\mathcal{F}_i$. Since $\mathcal{F}_i$ is a critical set, it then follows that the pair $(\mathbf{A}, \mathbf{C}_{\mathcal{Y}})$ is not detectable.

Thus, there exists an initial condition $\mathbf{x}[0] = \boldsymbol{\eta}$ that causes the measurement set $\mathbf{y}_{\mathcal{Y}}[k]$ corresponding to $\mathcal{Y}$ to be identically zero for all time, while the state $\mathbf{x}[k]$ remains bounded away from zero. The idea of the proof will be to demonstrate that the nodes in $\mathcal{Y}$ cannot distinguish between the zero initial condition and the initial condition $\boldsymbol{\eta}$ under an appropriately constructed attack. To this end, noting that each of the sets $\mathcal{H}_1$ and $\mathcal{H}_2$ are f-local and can hence act as valid adversarial sets, we consider the following executions σ and σ' of $\mathcal{T}$.

Execution σ: The initial condition is $\mathbf{x}[0] = \mathbf{0}$. The nodes in $\mathcal{H}_1$ are regular while the nodes in $\mathcal{H}_2$ are adversarial. The nodes in $\mathcal{H}_2$ pretend that their state estimates are $\hat{\mathbf{x}}_{\mathcal{H}_2}[k]$ and that their measurements are $\mathbf{C}_{\mathcal{H}_2}\mathbf{A}^k\boldsymbol{\eta}$, where $\hat{\mathbf{x}}_{\mathcal{H}_2}[k]$ represents the collection of the state estimates maintained by the nodes in $\mathcal{H}_2$ during the execution σ' of $\mathcal{T}$. Additionally, at each time-step, the nodes in $\mathcal{H}_2$ perform the exact same actions that they perform during the execution σ'.

Execution σ': The initial condition is $\mathbf{x}[0] = \boldsymbol{\eta}$. The nodes in $\mathcal{H}_2$ are regular while the nodes in $\mathcal{H}_1$ are adversarial. The nodes in $\mathcal{H}_1$ pretend that their state estimates are $\hat{\mathbf{x}}_{\mathcal{H}_1}[k]$ and that their measurements are zero, where $\hat{\mathbf{x}}_{\mathcal{H}_1}[k]$ represents the collection of the state estimates maintained by the nodes in $\mathcal{H}_1$ during the execution σ of $\mathcal{T}$. Additionally, at each time-step, the nodes in $\mathcal{H}_1$ perform the exact same actions that they perform during the execution σ.

Since the actions of the adversaries in the two executions described above are coupled, it becomes important to establish that such actions are in fact well-defined. To do so, we argue as follows. Consider the actions of the adversarial set $\mathcal{H}_2$ at time $k = 0$ of execution σ. Due to their omniscient nature, these adversaries can anticipate the information that a regular set $\mathcal{H}_2$ is supposed to transmit at time $k = 0$ of execution σ' based on algorithm $\mathcal{T}$. Thus, their actions are well-defined at time $k = 0$. Note that the last two statements rely on the deterministic nature of $\mathcal{T}$. Specifically, under a deterministic algorithm $\mathcal{T}$, the actions of the regular nodes are also deterministic, and hence, can be predicted in advance by an omniscient adversary who is aware of the information set available to such regular nodes. An identical argument defines the actions of the adversarial set $\mathcal{H}_1$ at time $k = 0$ of execution σ'. Since the actions of both the sets $\mathcal{H}_1$ and $\mathcal{H}_2$ at time $k = 0$ are well-defined in each of the executions σ and σ', the response of the regular nodes to such actions (in the respective executions) at time $k = 1$ can be anticipated by any adversarial set. Specifically, to generate their actions at time $k = 1$ of execution σ (resp., execution σ'), the adversarial set $\mathcal{H}_2$ (resp., $\mathcal{H}_1$) simply simulates execution σ' (resp., execution σ) for time $k = 0$ to figure out how a regular set $\mathcal{H}_2$ (resp., $\mathcal{H}_1$) would act at time $k = 1$ of execution σ' (resp., execution σ). Repeating the above argument reveals that the actions of the respective adversarial sets in each of the executions σ and σ' are well-defined at every time step.

Based on the attack described above, it is clear that the nodes in $\mathcal{Y}$ receive the same state estimate and measurement information from the nodes in $\mathcal{H}$ in each of the two executions. Further, their own measurements are identically zero for all time in each of the two executions. Hence, based on such identical information, it is impossible for the nodes in $\mathcal{Y}$ to resolve the difference in the underlying initial conditions via algorithm $\mathcal{T}$. This leads to the desired contradiction and completes the proof. ■

Remark 3.3.2 *Interestingly, the necessary condition presented in the above theorem bears close resemblance to the necessary condition in [92,100] for resilient broadcasting subject to the same f-local Byzantine adversary model that we consider here. This similarity can be attributed to the following analogy: viewing the virtual nodes as originators of messages in a broadcasting context, Problem 3 can be interpreted as a version of the resilient broadcasting problem where the regular nodes are required to agree (asymptotically) on a time-varying message that captures the state evolution of the system.*

Our next result provides a necessary condition for an f-total (and hence, also an f-local) adversarial model.

Theorem 3.3.3 *Suppose there exists a causal, synchronous and deterministic algorithm that solves the variant of Problem 3 corresponding to an f-total Byzantine adversary model. Then, the following equivalent statements are true.*

(i) Consider any minimal critical set $\mathcal{F}_i \in \mathcal{M}$. There exists no f-total pair cut w.r.t. s_i.

(ii) Consider a node $i \in \mathcal{V}$ such that $(\mathbf{A}, \mathbf{C}_i)$ is not detectable. Let $\mathcal{X}_i$ denote the set of all nodes in $\mathcal{G}$ that have directed paths to node i, and consider a set $\mathcal{D}_i \subseteq \mathcal{X}_i$ such that $|\mathcal{D}_i| \leq 2f$. Let $\mathcal{P}_i \subseteq \mathcal{X}_i$ represent the set of nodes that have directed paths to node i in the induced subgraph obtained by removing $\mathcal{D}_i$ from $\mathcal{G}$. Then, $(\mathbf{A}, \mathbf{C}_{i \cup \mathcal{P}_i})$ is detectable.

The proof of necessity mimics the proof of Theorem 3.3.1, while the equivalence between the two conditions stated in Theorem 3.3.3 is established in Section 3.10.1.

Remark 3.3.4 *In [82, 83], the authors showed that for centralized systems subject to f sensor attacks, a necessary condition for estimating the state asymptotically is that the system should remain detectable after the removal of any $2f$ sensors. In our present distributed setting, the maximum information about the state that any given node i can hope to obtain is from the set $\{i \cup \mathcal{X}_i\}$, where $\mathcal{X}_i$ is defined as in Theorem 3.3.3. Thus, the second part of Theorem 3.3.3 generalizes the necessary conditions in [82,83]. In [70, 101], the authors established that the graph-connectivity metric plays a pivotal role in the analysis of fault-tolerant and resilient distributed consensus algorithms for settings where there are no underlying state dynamics that need to be estimated. The results stated in Theorems 3.3.1 and 3.3.3 differ from those in [70, 101] since they blend both graph-theoretic and system-theoretic requirements. Finally, it can be easily shown that when there are no adversaries, i.e., when $f = 0$, the conditions identified in Theorem 3.3.3 reduce to the necessary and sufficient condition for distributed state estimation, namely every source component (strong components with no incoming edges) of the graph should be detectable [1, 30, 57, 58, 102].*

We now discuss certain implications of Theorem 3.3.3. Given an LTI system (2.1), a measurement model specified by (2.2), and a communication graph $\mathcal{G}$, it is of both theoretical and practical interest to know the maximum number of adversaries that can be tolerated when one seeks to solve Problem 3. Leveraging Theorem 3.3.3, we can provide an upper bound on this number, as follows.

Corollary 3.3.5 *Let k denote the smallest positive integer such that there exists a k-total pair cut w.r.t. s_i for some $\mathcal{F}_i \in \mathcal{M}$. Then, the total number of adversaries f must satisfy the inequality $f < k$ for Problem 3 to have a solution.*[2]

Corollary 3.3.6 *The condition $|\mathcal{F}_i| \geq (2f+1)\ \forall \mathcal{F}_i \in \mathcal{M}$ is necessary for resilient distributed state estimation subject to the f-local or f-total adversarial model.*

The proof of the above result is straightforward and is hence omitted here. With the above corollary in hand, one can gain insights regarding the distribution of certain specific critical sets in the network. To do so, given an eigenvalue $\lambda_j \in \Lambda_U(\mathbf{A})$, let $\{\boldsymbol{\rho}_1^{(j)}, \cdots, \boldsymbol{\rho}_{g_{\mathbf{A}}(\lambda_j)}^{(j)}\}$ represent a basis for the null space of $(\mathbf{A} - \lambda_j \mathbf{I}_n)$, and let $\phi_i^{(j)} = span\{\boldsymbol{\rho}_i^{(j)}\}, i \in \{1, \cdots, g_{\mathbf{A}}(\lambda_j)\}$. We say that node i can detect the subspace $\phi_i^{(j)}$ if $\mathbf{C}_i\boldsymbol{\rho}_i^{(j)} \neq \mathbf{0}$.[3] Let $\mathcal{W}_i^{(j)} \subseteq \mathcal{V}$ denote the set of all nodes that can detect $\phi_i^{(j)}$. The next result then readily follows from Corollary 3.3.6 and the classical PBH test [104].

Proposition 3.3.1 *For each $\lambda_j \in \Lambda_U(\mathbf{A})$, if $\mathcal{W}_i^{(j)} \subset \mathcal{V}$, where $1 \leq i \leq g_{\mathbf{A}}(\lambda_j)$, then $|\mathcal{W}_i^{(j)}| \geq (2f+1)$ is a necessary condition for resilient distributed state estimation subject to the f-local or f-total adversarial model.*

For systems with distinct eigenvalues, a direct consequence of the above result is the requirement of at least $(2f+1)$ nodes that can detect each unstable eigenvalue of the system. The preceding analysis builds up to the distributed estimation strategy

[2] Similar bounds for static power system models subject to attacks were obtained in [103].

[3] Throughout the chapter, for the sake of conciseness, we use the terminology "node i can detect eigenvalue λ_j" to imply that $rank \begin{bmatrix} \mathbf{A}-\lambda_j\mathbf{I}_n \\ \mathbf{C}_i \end{bmatrix} = n$. Each stable eigenvalue is considered detectable w.r.t. the measurements of every node.

adopted in this chapter. In particular, our approach involves identifying the locally detectable and undetectable eigenvalues associated with a given node, and subsequently devising separate estimation strategies for the subspaces associated with such eigenvalues. We formalize this idea in the next section.

Remark 3.3.7 *Two important directions of future investigation are (i) finding an efficient algorithm (if one exists) for computing k in Corollary 3.3.5 either exactly or approximately, and (ii) determining whether the conditions stated in Theorem 3.3.1 (resp., Theorem 3.3.3) are sufficient for achieving resilient distributed state estimation subject to an f-local (resp., f-total) Byzantine adversary model. Note that the main source of computational complexity associated with the first point lies in finding all the minimal critical sets associated with the given system.*

3.4 Resilient Distributed State Estimation

3.4.1 Preliminaries

For each eigenvalue $\lambda \in sp(\mathbf{A})$, let $\mathbf{V}(\lambda)$ represent a block diagonal matrix with the Jordan blocks corresponding to λ (in the standard Jordan canonical representation of $\mathbf{A}$) along the main block diagonal. We begin by recalling certain properties of the real Jordan canonical form of a square matrix that will be useful for our subsequent development [105]. We first note that if λ represents a non-real eigenvalue of $\mathbf{A}$ and $\bar{\lambda}$ represents its complex-conjugate, then [105, Lemma 3.1.18] ensures that λ and $\bar{\lambda}$ have the same Jordan structure. Next, let $\lambda = a + ib$ where $a, b \in \mathbb{R}$, and $i = \sqrt{-1}$. Let $\mathbf{D}(a,b)$ be defined as $\mathbf{D}(a,b) \triangleq \begin{bmatrix} a & b \\ -b & a \end{bmatrix}$. Then, the matrix $diag(\mathbf{V}(\lambda), \mathbf{V}(\bar{\lambda}))$ is similar to a real block upper triangular matrix $\mathbf{W}(\lambda) \in \mathbb{R}^{2a_{\mathbf{A}}(\lambda) \times 2a_{\mathbf{A}}(\lambda)}$ which has $a_{\mathbf{A}}(\lambda)$ 2-by-2 blocks $\mathbf{D}(a,b)$ on the main block diagonal and $(a_{\mathbf{A}}(\lambda)-1)$ blocks $\mathbf{I}_2$ on the block super-diagonal. Henceforth, for a non-real eigenvalue $\lambda \in sp(\mathbf{A})$, $\mathbf{W}(\lambda)$ will have the meaning discussed above. Let $sp(\mathbf{A}) = \{\{\lambda_1, \bar{\lambda}_1\}, \cdots, \{\lambda_p, \bar{\lambda}_p\}, \lambda_{p+1}, \cdots, \lambda_\gamma\}$ with the first p pairs representing the non-real eigenvalues, and λ_{p+1} to λ_γ representing the real eigenvalues of $\mathbf{A}$. Then, the *real Jordan canonical form theorem* [105, Theorem 3.4.1.5] can be stated as follows.

Theorem 3.4.1 *There exists a real similarity transformation matrix* $\mathbf{T}$ *that transforms the state transition matrix* $\mathbf{A}$ *in* (2.1) *to a real block diagonal matrix* $\mathbf{M}$ *given by* $\mathbf{M} = diag(\mathbf{W}(\lambda_1), \cdots, \mathbf{W}(\lambda_p), \mathbf{V}(\lambda_{p+1}), \cdots, \mathbf{V}(\lambda_\gamma))$.

With $\mathbf{T}$ as in the above theorem, and $\mathbf{z}[k] = \mathbf{T}^{-1}\mathbf{x}[k]$, the dynamics (2.1) are transformed into the form

$$\begin{aligned} \mathbf{z}[k+1] &= \mathbf{M}\mathbf{z}[k] \\ \mathbf{y}_i[k] &= \bar{\mathbf{C}}_i\mathbf{z}[k], \quad \forall i \in \{1, \cdots, N\} \end{aligned} \tag{3.1}$$

where $\mathbf{M} = \mathbf{T}^{-1}\mathbf{A}\mathbf{T}$ and $\bar{\mathbf{C}}_i = \mathbf{C}_i\mathbf{T}$. For a non-real eigenvalue pair $\{\lambda_j, \bar{\lambda}_j\} \in sp(\mathbf{A})$, let $\mathbf{z}^{(j)}[k] \in \mathbb{R}^{2a_{\mathbf{A}}(\lambda_j)}$ represent the portion of the state $\mathbf{z}[k]$ associated with the matrix $\mathbf{W}(\lambda_j)$. Similarly, for a real eigenvalue $\lambda_j \in sp(\mathbf{A})$, $\mathbf{z}^{(j)}[k] \in \mathbb{R}^{a_{\mathbf{A}}(\lambda_j)}$ is the portion of the state $\mathbf{z}[k]$ associated with the matrix $\mathbf{V}(\lambda_j)$. For each node i, we denote the detectable and undetectable eigenvalues by the sets $\mathcal{O}_i$ and $\overline{\mathcal{O}}_i$, respectively. Next, we introduce the notion of *source nodes*.

Definition 3.4.1 *(**Source nodes**) For each* $\lambda_j \in \Lambda_U(\mathbf{A})$*, the set of nodes that can detect* λ_j *is denoted by* $\mathcal{S}_j$*, and called the set of source nodes for* λ_j*.*

We now proceed to develop an estimation scheme that enables each regular node to estimate $\mathbf{z}[k]$ (from which they can obtain $\mathbf{x}[k] = \mathbf{T}\mathbf{z}[k]$). Accordingly, let $\hat{\mathbf{z}}_i^{(j)}[k]$ denote the estimate of $\mathbf{z}^{(j)}[k]$ (the portion of $\mathbf{z}[k]$ corresponding to the eigenvalue λ_j[4]) maintained by node $i \in \mathcal{R}$. For each $\lambda_j \in \Lambda_U(\mathbf{A})$, our estimation scheme relies on separate strategies for nodes in $\mathcal{S}_j$ and $\mathcal{V} \setminus \mathcal{S}_j$. In particular, each node in $\mathcal{S}_j$ employs a Luenberger observer for estimating $\mathbf{z}^{(j)}[k]$. The nodes in $\mathcal{V} \setminus \mathcal{S}_j$, on the other hand, cannot detect the eigenvalue λ_j, and thus rely on a resilient consensus algorithm to estimate $\mathbf{z}^{(j)}[k]$. In what follows, we discuss these ideas in detail.

[4] Throughout the rest of the chapter, the terminology "$\mathbf{z}^{(j)}[k]$ corresponds to the eigenvalue λ_j" should be interpreted as $\mathbf{z}^{(j)}[k]$ corresponds to the eigenvalue pair $\{\lambda_j, \bar{\lambda}_j\}$ for a non-real eigenvalue $\lambda_j \in sp(\mathbf{A})$.

The first step in the estimation process involves the above common coordinate transformation given by $\mathbf{z}[k] = \mathbf{T}^{-1}\mathbf{x}[k]$, to be performed by each regular node of the graph. As this only requires knowledge of the system matrix $\mathbf{A}$ (which is assumed to be known by all the nodes), all of the nodes can do this in a distributed manner (e.g., by using an agreed-upon convention for ordering the eigenvalues and corresponding eigenvectors). Building on the general theme in [57], we first present a method that allows a regular node $i \in \mathcal{R}$ to estimate the locally detectable portion of the state $\mathbf{z}[k]$ *without* communicating with neighbors. To this end, consider the following result.

Lemma 3.4.2 *Let $\lambda_j \in \mathcal{O}_i$ be a non-real eigenvalue. Let $\bar{\mathbf{C}}_i^{(j)}$ denote the columns of $\bar{\mathbf{C}}_i$ corresponding to $\mathbf{W}(\lambda_j)$ in (3.1). Then, the pair $(\mathbf{W}(\lambda_j), \bar{\mathbf{C}}_i^{(j)})$ is detectable.*

Proof We claim that $\lambda_j \in \mathcal{O}_i$ if and only if $\bar{\lambda}_j \in \mathcal{O}_i$. It suffices to prove necessity since the proof for sufficiency will follow an identical argument. We prove necessity by contradiction. Suppose node i can detect λ_j (i.e., $\lambda_j \in \mathcal{O}_i$), but cannot detect $\bar{\lambda}_j$. Then, there exists $\mathbf{v} \neq \mathbf{0}$ such that $\mathbf{A}\mathbf{v} = \bar{\lambda}_j\mathbf{v}$ and $\mathbf{C}_i\mathbf{v} = \mathbf{0}$. Taking complex conjugates on both sides of these equations reveals that node i cannot detect λ_j, leading to the desired contradiction. Given that a similarity transformation maps $(\mathbf{A}, \mathbf{C}_i)$ to $(\mathbf{M}, \bar{\mathbf{C}}_i)$, it then follows that $\{\lambda_j, \bar{\lambda}_j\}$ are detectable eigenvalues w.r.t. $(\mathbf{M}, \bar{\mathbf{C}}_i)$. Detectability of the pair $(\mathbf{W}(\lambda_j), \bar{\mathbf{C}}_i^{(j)})$ then follows readily from the PBH test by noting the structure of the matrix $\mathbf{M}$. ■

Let $\mathcal{O}_i = \{\{\lambda_{n_1}, \bar{\lambda}_{n_1}\}, \cdots, \{\lambda_{n_{p_i}}, \bar{\lambda}_{n_{p_i}}\}, \lambda_{n_{p_i+1}}, \cdots, \lambda_{n_{\gamma_i}}\}$, where the first p_i pairs represent the non-real eigenvalues, and $\lambda_{n_{p_i+1}}$ to $\lambda_{n_{\gamma_i}}$ represent the real eigenvalues of $\mathbf{A}$ that are detectable w.r.t. the measurements of node i. Let $\mathbf{M}_{\mathcal{O}_i} = diag(\mathbf{W}(\lambda_{n_1}), \cdots, \mathbf{W}(\lambda_{n_{p_i}}), \mathbf{V}(\lambda_{n_{p_i+1}}), \cdots, \mathbf{V}(\lambda_{n_{\gamma_i}}))$. Let $\mathbf{C}_{\mathcal{O}_i}$ represent the columns of $\bar{\mathbf{C}}_i$ corresponding to the matrix $\mathbf{M}_{\mathcal{O}_i}$, and $\mathbf{z}_{\mathcal{O}_i}[k]$ denote the portion of the state $\mathbf{z}[k]$ corresponding to the detectable eigenvalues of node i, i.e., corresponding to $\mathcal{O}_i$. Based on Lemma 3.4.2, it is easy to see that the pair $(\mathbf{M}_{\mathcal{O}_i}, \mathbf{C}_{\mathcal{O}_i})$ is detectable. Thus, a standard Luenberger observer can be locally constructed by node i for estimating $\mathbf{z}_{\mathcal{O}_i}[k]$. The details of such a construction are straightforward, and are similar to those in Section VI-A of [58]. We thus skip minor details and state the following result which will be useful later on.

Lemma 3.4.3 *For each regular node $i \in \mathcal{R}$ and each $\lambda_j \in \mathcal{O}_i$, a Luenberger observer can be locally constructed by node i such that* $\lim_{k\to\infty} \|\hat{\mathbf{z}}_i^{(j)}[k] - \mathbf{z}^{(j)}[k]\| = 0$.

Based on the previous result, we see that a regular node i can estimate certain portions of the state space without having to exchange information with neighbors. The challenge, however, lies in estimating the locally undetectable portion of the state in the presence of adversaries. The following section presents a resilient consensus based strategy to address this issue.

3.4.2 Local-Filtering Based Resilient Estimation

For any $\lambda_j \in sp(\mathbf{A})$, let $z^{(jm)}[k]$ denote the m-th component of the vector $\mathbf{z}^{(j)}[k]$, and let $\hat{z}_i^{(jm)}[k]$ denote the estimate of that component maintained by node $i \in \mathcal{V}$. Consider an unstable eigenvalue $\lambda_j \in \overline{\mathcal{O}}_i$. For such an eigenvalue, node i has to rely on the information received from its neighbors, some of which might be adversarial, in order to estimate $\mathbf{z}^{(j)}[k]$. To this end, we propose a resilient consensus algorithm that requires each regular node $i \in \mathcal{V} \setminus \mathcal{S}_j$ to update its estimate of $\mathbf{z}^{(j)}[k]$ using the following two stage filtering strategy:

1) At each time-step k, each regular node i collects the state estimates of $\mathbf{z}^{(j)}[k]$ received from *only* those neighbors that belong to a certain subset $\mathcal{N}_i^{(j)} \subseteq \mathcal{N}_i$ (to be defined later). For every component m of $\mathbf{z}^{(j)}[k]$, the estimates of $z^{(jm)}[k]$ received from nodes in $\mathcal{N}_i^{(j)}$ are sorted from largest to smallest.

2) For each component m of $\mathbf{z}^{(j)}[k]$, node i removes the largest and smallest f estimates (i.e., removes $2f$ estimates in all) of $z^{(jm)}[k]$ received from nodes in $\mathcal{N}_i^{(j)}$, and computes the quantity:

$$\bar{z}_i^{(jm)}[k] = \sum_{l \in \mathcal{M}_i^{(jm)}[k]} w_{il}^{(jm)}[k] \hat{z}_l^{(jm)}[k], \tag{3.2}$$

where $\mathcal{M}_i^{(jm)}[k] \subset \mathcal{N}_i^{(j)}(\subseteq \mathcal{N}_i)$ is the set of nodes from which node i chooses to accept estimates of $z^{(jm)}[k]$ at time-step k, after removing the f largest and f smallest estimates of $z^{(jm)}[k]$ from $\mathcal{N}_i^{(j)}$. Node i assigns the weight $w_{il}^{(jm)}[k]$ to the l-th node at the k-th time-step for estimating the m-th component of $\mathbf{z}^{(j)}[k]$. The weights are nonnegative and chosen to satisfy $\sum_{l\in\mathcal{M}_i^{(jm)}[k]} w_{il}^{(jm)}[k] = 1, \forall \lambda_j \in \overline{\mathcal{O}}_i$ and for each component m of $\mathbf{z}^{(j)}[k]$. With the quantities $\bar{z}_i^{(jm)}[k]$ in hand, node i updates $\hat{\mathbf{z}}_i^{(j)}[k]$ as follows:

$$\hat{\mathbf{z}}_i^{(j)}[k+1] = \begin{cases} \mathbf{V}(\lambda_j)\bar{\mathbf{z}}_i^{(j)}[k], & \text{if } \lambda_j \in \overline{\mathcal{O}}_i \text{ is real} \\ \mathbf{W}(\lambda_j)\bar{\mathbf{z}}_i^{(j)}[k], & \text{if } \lambda_j \in \overline{\mathcal{O}}_i \text{ is not real,} \end{cases} \tag{3.3}$$

where $\bar{\mathbf{z}}_i^{(j)}[k] = \left[\bar{z}_i^{(j1)}[k], \cdots, \bar{z}_i^{(j\sigma_j)}[k]\right]^T$, $\sigma_j = a_{\mathbf{A}}(\lambda_j)$ if $\lambda_j \in \overline{\mathcal{O}}_i$ is real, and $\sigma_j = 2a_{\mathbf{A}}(\lambda_j)$ if $\lambda_j \in \overline{\mathcal{O}}_i$ is not real.

We refer to the above algorithm as the Local-Filtering based Resilient Estimation (LFRE) algorithm. For implementing this algorithm, a regular node i needs to construct the set $\mathcal{N}_i^{(j)}$, $\forall \lambda_j \in \overline{\mathcal{O}}_i$, based on the relative positions of its neighbors (with respect to its own position) in $\mathcal{G}$. We will provide the exact definition of $\mathcal{N}_i^{(j)}$, and a distributed algorithm for constructing such a set in the following sections where we analyze the convergence of the LFRE algorithm. We conclude this section by commenting on certain features of the LFRE algorithm.

Remark 3.4.4 *The rationale behind performing a real Jordan canonical decomposition at every node (as opposed to a standard Jordan transformation) is to ensure that the state estimates featuring in equations (3.2) and (3.3) are real at every time-step, thereby making the sorting operation performed in Step 1 of the algorithm meaningful. At any time-step, if a regular node i either receives a non-real estimate of $z^{(jm)}[k]$ from some node $l \in \mathcal{N}_i^{(j)}$ or does not receive an estimate at all, it would immediately identify node l as an adversarial node, and simply assign a 0 value to node l's estimate of $z^{(jm)}[k]$. Note that every regular node in $\mathcal{N}_i^{(j)}$ will always transmit a real estimate to node i at every time-step.*

Remark 3.4.5 *The strategy of disregarding the most extreme values in one's neighborhood, and using a convex combination of the rest for performing linear scalar updates, has been used for designing resilient distributed algorithms for consensus [87, 88, 106] and optimization [89, 90] problems. In this chapter, we show that such algorithms can also be used for resilient distributed state estimation, with certain substantial differences arising from the fact that the nodes are trying to track the state of an external dynamical system.*

Remark 3.4.6 *The consensus weights $w_{il}^{(jm)}$ appearing in equation (3.2) can be chosen arbitrarily to achieve an exponential rate of convergence, as long as the weights meet the rules specified by the LFRE algorithm. Since our primary focus is on resilience against worst-case adversarial behavior, the problem of optimizing such weights (or exploiting sensor memory) for achieving improved performance against noise is not considered in this chapter. In a non-adversarial setting (i.e., when $f = 0$), the proposed LFRE algorithm will continue to guarantee exponential convergence in the absence of noise, and bounded mean square error in the presence of i.i.d. noise with bounded second moments (provided the topological conditions outlined in Section 3.7 are met). However, disregarding the estimates of certain neighbors in the absence of attacks may potentially degrade performance against noise; we do not delve deeper into this topic here.*

It should be noted that the algorithmic development in this section can be considerably simplified if more structure is imposed on the system matrix $\mathbf{A}$ (for instance, the assumption made in [84] that $\mathbf{A}$ has only real, distinct eigenvalues).

3.5 Analysis of the Resilient Distributed Estimation Strategy

In this section, we provide our main result concerning the convergence of the LFRE algorithm. Let $\Omega_U(\mathbf{A}) \triangleq \{\lambda_j \in \Lambda_U(\mathbf{A}) | \mathcal{V} \setminus \mathcal{S}_j \text{ is non-empty}\}$. By this definition, all nodes are source nodes for each eigenvalue in $\Lambda_U(\mathbf{A}) \setminus \Omega_U(\mathbf{A})$, and are hence capable of recovering the corresponding portions of the state based on locally

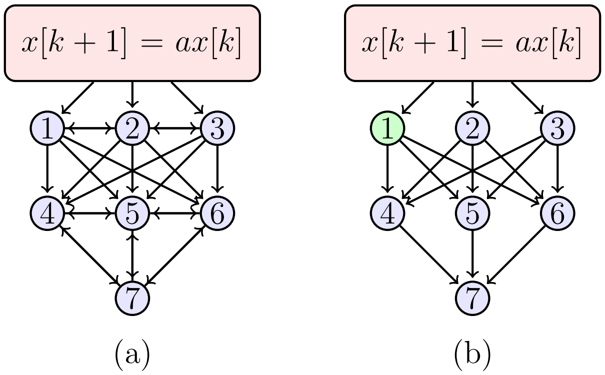


Fig. 3.2. A scalar unstable plant is monitored by a network of 7 nodes as depicted by the figure on the left. Nodes 1, 2 and 3 are the source nodes for this system. The figure on the right represents a subgraph of the original graph satisfying the properties of a MEDAG in Definition 3.5.1 for all 1-local sets (i.e., with $f = 1$). For example, when $\mathcal{A} = \{1\}$ (as shown in the right figure), every non-source node has at least $2f + 1 = 3$ neighbors. The levels that partition $\mathcal{R} = \mathcal{V} \setminus \mathcal{A}$ are level 0 with nodes 2 and 3, level 1 with nodes 4, 5 and 6, and level 2 with node 7. Each regular node has all its regular neighbors in levels that are numbered lower than its own.

constructed Luenberger observers (as discussed in Section 3.4.1). Consequently, the LFRE algorithm specifically applies to only those eigenvalues that belong to $\Omega_U(\mathbf{A})$. Consider the following definition.

Definition 3.5.1 ***(Mode Estimation Directed Acyclic Graph (MEDAG))*** *Consider an eigenvalue $\lambda_j \in \Omega_U(\mathbf{A})$. Suppose there exists a spanning subgraph $\mathcal{G}_j = (\mathcal{V}, \mathcal{E}_j)$ of $\mathcal{G}$ with the following properties for all f-local sets $\mathcal{A}$ and $\mathcal{R} = \mathcal{V} \setminus \mathcal{A}$.*

(i) If $i \in \{\mathcal{V} \setminus \mathcal{S}_j\} \cap \mathcal{R}$, then $|\mathcal{N}_i^{(j)}| \geq 2f + 1$, where $\mathcal{N}_i^{(j)} = \{l | (l, i) \in \mathcal{E}_j\}$ represents the neighborhood of node i in $\mathcal{G}_j$.

(ii) There exists a partition of $\mathcal{R}$ into the sets $\{\mathcal{L}_0^{(j)}, \cdots, \mathcal{L}_{T_j}^{(j)}\}$, where $T_j \in \mathbb{N}_+$, $\mathcal{L}_0^{(j)} = \mathcal{S}_j \cap \mathcal{R}$, and if $i \in \mathcal{L}_q^{(j)}$ (where $1 \leq q \leq T_j$), then $\mathcal{N}_i^{(j)} \cap \mathcal{R} \subseteq \bigcup_{r=0}^{q-1} \mathcal{L}_r^{(j)}$. Furthermore, $\mathcal{N}_i^{(j)} = \emptyset, \forall i \in \mathcal{L}_0^{(j)}$.

Then, we call $\mathcal{G}_j$ a Mode Estimation Directed Acyclic Graph (MEDAG) for $\lambda_j \in \Omega_U(\mathbf{A})$.

An example of a MEDAG is shown in Figure 3.2. The "for all $\mathcal{A}$" in the definition accounts for the fact that the set of adversarial nodes during the process of state estimation is unknown, and hence can be any f-local set of $\mathcal{V}$. Note that T_j and the levels $\mathcal{L}_0^{(j)}$ to $\mathcal{L}_{T_j}^{(j)}$ can vary across different f-local sets. For a given f-local set $\mathcal{A}$, we say a regular node $i \in \mathcal{L}_m^{(j)}$ "belongs to level m", where the levels are indicative of the distances of the regular nodes from the source set $\mathcal{S}_j$. The first property indicates that every regular node $i \in \mathcal{V} \setminus \mathcal{S}_j$ has at least $(2f+1)$ neighbors in the subgraph $\mathcal{G}_j$, while the second property indicates that all its regular neighbors in such a subgraph belong to levels strictly preceding its own level. In essence, the edges of the MEDAG $\mathcal{G}_j$ represent a medium for transmitting information securely from the source nodes $\mathcal{S}_j$ to the non-source nodes, by preventing the adversaries from forming a bottleneck between such nodes. Intuitively, this requires redundant nodes and edges, and such a requirement is met by the first property of the MEDAG. In particular, as regards measurement redundancy, it follows from the definition that for each $\lambda_j \in \Omega_U(\mathbf{A})$, a MEDAG $\mathcal{G}_j$ contains at least $(2f+1)$ source nodes that can detect λ_j.[5] The LFRE algorithm described in the previous section relies on a special *uni-directional* information flow pattern that requires a node i to listen to *only* its neighbors in $\mathcal{N}_i^{(j)}$ for estimating $\mathbf{z}^{(j)}[k]$. The second property of a MEDAG then indicates that nodes in level m only use the estimates of regular nodes in levels 0 to $m-1$ for recovering $\mathbf{z}^{(j)}[k]$. The implications of the above properties will become apparent in the proof of the following result which provides a sufficient condition for solving Problem 3 based on our approach.

Theorem 3.5.1 *Suppose that $\mathcal{G}$ contains a MEDAG $\mathcal{G}_j$ for each $\lambda_j \in \Omega_U(\mathbf{A})$. Then, based on the LFRE dynamics described by equations (3.2) and (3.3), each regular*

[5] Recall from the discussion immediately following Proposition 3.3.1 that such a condition is in fact necessary for systems with distinct eigenvalues.

node $i \in \mathcal{R}$ can asymptotically estimate the state of the plant, despite the actions of any f-locally bounded set of Byzantine adversaries.

The proof of the above theorem is given in Section 3.10.2. Notice that Theorem 3.5.1 hinges on the existence of a MEDAG $\mathcal{G}_j$, for each $\lambda_j \in \Omega_U(\mathbf{A})$; in the following section we describe an approach for checking whether a given graph $\mathcal{G}$ contains such MEDAGs.

3.6 Checking the Existence of a MEDAG

From the foregoing discussion, it is apparent that the MEDAGs described in Definition 3.5.1 play a key role in solving Problem 3 based on our proposed technique. In particular, recall that for each $\lambda_j \in \Omega_U(\mathbf{A})$, the LFRE algorithm described in Section 3.4.2 requires a regular node $i \in \mathcal{V} \setminus \mathcal{S}_j$ to accept estimates from only its neighbor set $\mathcal{N}_i^{(j)}$ in the MEDAG $\mathcal{G}_j$ for estimating $\mathbf{z}^{(j)}[k]$. With these points in mind, our immediate goal in this section will be to develop a distributed algorithm, namely Algorithm 1, that constructs a MEDAG $\mathcal{G}_j$ for each $\lambda_j \in \Omega_U(\mathbf{A})$, and in the process enables each regular node i to determine the set $\mathcal{N}_i^{(j)}$ for each $\lambda_j \in \overline{\mathcal{O}}_i$. The construction of these MEDAGs constitutes the initialization phase of our design, which can then be followed up by the LFRE algorithm described earlier. We briefly describe the implementation of Algorithm 1 as follows.

Algorithm 1 requires each node i to maintain a counter $c_i(j)$ and a list of indices $\mathcal{N}_i^{(j)}$ for each $\lambda_j \in \Omega_U(\mathbf{A})$. The nodes in $\mathcal{N}_i^{(j)} \subseteq \mathcal{N}_i$ will be the parents of node i in the DAG constructed for the estimation of $\mathbf{z}^{(j)}[k]$. Algorithm 1 is initialized with $c_i(j) = 0$ and $\mathcal{N}_i^{(j)} = \emptyset$, for each $i \in \mathcal{V}$. Subsequently, the algorithm proceeds in rounds where in the first round each node in $\mathcal{S}_j$ broadcasts the message "1" to its out-neighbors, sets $c_i(j) = 1$, maintains $\mathcal{N}_i^{(j)} = \emptyset$ for all future rounds, and goes to sleep. Each node $i \in \mathcal{V} \setminus \mathcal{S}_j$ waits until it has received "1" from at least $(2f+1)$ distinct neighbors, at which point it sets $c_i(j) = 1$, appends the labels of each of the neighbors from which it received "1" to $\mathcal{N}_i^{(j)}$, broadcasts the message "1" to its

Algorithm 1 MEDAG Construction Algorithm

```
For each eigenvalue $\lambda_j \in \Omega_U(\mathbf{A})$ **do**:
**Initialization**: Initialize $c_i(j) = 0$, $\mathcal{N}_i^{(j)} = \emptyset$, $\forall i \in \mathcal{V}$. Each node determines whether it belongs to $\mathcal{S}_j$.
**Actions of the source nodes**: Each node in $\mathcal{S}_j$ updates its counter value $c_i(j) = 1$, and transmits the message "1" to its out-neighbors. Following this step, it does not listen to any other node, i.e., $\mathcal{N}_i^{(j)} = \emptyset$ and $c_i(j) = 1$, $\forall i \in \mathcal{S}_j$ for the remainder of the algorithm.
**Actions of the non-source nodes**: Each node $i \in \mathcal{V} \setminus \mathcal{S}_j$ does the following:
    • If $c_i(j) = 0$ *and* node $i$ has received "1" from at least $(2f + 1)$ distinct neighbors (not necessarily all in the same round), it updates $c_i(j)$ to 1, appends the labels of the neighbors from which it received "1" to $\mathcal{N}_i^{(j)}$, and transmits "1" to its out-neighbors.
    • If $c_i(j) = 1$, it discards all messages received from its neighbors, i.e., it does not update $c_i(j)$ or $\mathcal{N}_i^{(j)}$.
**Return** : A set of sets $\{\mathcal{N}_i^{(j)}\}$, $\lambda_j \in \Omega_U(\mathbf{A})$, $i \in \mathcal{V}$.
```

out-neighbors, and goes to sleep. Let $\mathcal{R}' \subseteq \mathcal{V}$ denote the set of nodes that behave regularly during the execution of Algorithm 1. We say that the MEDAG construction algorithm "*terminates for* λ_j" if there exists $T_j \in \mathbb{N}_+$ such that $c_i(j) = 1$ $\forall i \in \mathcal{R}'$, for all rounds following round T_j. The **objective** of the algorithm is to return a set of sets $\{\mathcal{N}_i^{(j)}\}$, where $\lambda_j \in \Omega_U(\mathbf{A})$, and $i \in \mathcal{V}$.

We emphasize that in addition to misbehavior during the state estimation phase (run-time), an adversarial node is allowed to misbehave during the implementation of Algorithm 1 (design-time) as well. For example, it can transmit the message out of turn, i.e., before receiving "1" from at least $(2f + 1)$ neighbors. It can also choose not to transmit the message at all. Note however that we must have $\mathcal{V} \setminus \mathcal{R}' \subseteq \mathcal{A}$, i.e., the f-local set of adversaries during the estimation phase must contain the set of adversaries during the design phase. In the next section, we shall detail graph conditions that guarantee the termination of the MEDAG construction algorithm under arbitrary adversarial behavior. For the following discussion, we characterize the properties of the output of Algorithm 1 if it terminates. To this end, consider the spanning subgraph $\mathcal{G}_j = (\mathcal{V}, \mathcal{E}_j)$ induced by the sets $\{\mathcal{N}_i^{(j)}\}$ returned by Algorithm 1. Keeping in mind that $\mathcal{R}' \supseteq \mathcal{R}$ represents the set of nodes that behave regularly during the execution of Algorithm 1, we have the following results; proofs of these results can be found in [84].

Proposition 3.6.1 *Suppose Algorithm 1 terminates for some $\lambda_j \in \Omega_U(\mathbf{A})$, and returns the sets $\{\mathcal{N}_i^{(j)}\}$. Then, the spanning subgraph $\mathcal{G}_j$ induced by these sets contains no directed cycles where every node belongs to $\mathcal{R}'$.*

Let $\mathcal{L}_{m-1}^{(j)}$ denote the set of all nodes in $\mathcal{R}'$ that update their counter value from 0 to 1 in round m of Algorithm 1, i.e., $\mathcal{L}_0^{(j)} = \mathcal{S}_j \cap \mathcal{R}'$, and so on.[6]

Proposition 3.6.2 *Suppose Algorithm 1 terminates for some $\lambda_j \in \Omega_U(\mathbf{A})$. Let T_j denote the smallest integer such that in round T_j, $c_i(j) = 1 \ \forall i \in \mathcal{R}'$. Then, the sets $\{\mathcal{L}_0^{(j)}, \cdots, \mathcal{L}_{T_j}^{(j)}\}$ form a partition of the set $\mathcal{R}'$ in $\mathcal{G}_j$.*

Theorem 3.6.1 *Suppose the MEDAG construction algorithm terminates for $\lambda_j \in \Omega_U(\mathbf{A})$. Then, there exists a subgraph $\mathcal{G}_j$ satisfying the properties of a MEDAG for all f-local sets $\mathcal{A}$ that contain $\mathcal{V} \setminus \mathcal{R}'$ as a subset.*

Proof The result follows immediately from Propositions 3.6.1 and 3.6.2. ■

Remark 3.6.2 *Based on the above theorem, we make the following observations. If Algorithm 1 terminates for each $\lambda_j \in \Omega_U(\mathbf{A})$, and $\mathcal{V} \setminus \mathcal{R}' = \emptyset$, then the $\mathcal{G}_j$ subgraphs satisfy all the properties of a MEDAG and we can directly invoke Theorem 3.5.1. If Algorithm 1 terminates for each $\lambda_j \in \Omega_U(\mathbf{A})$ and $\mathcal{V} \setminus \mathcal{R}' \neq \emptyset$, (i.e., there is some adversarial activity during the MEDAG construction phase), then we do not need to*

[6] Note that the method developed in this chapter allows even some of the source nodes in $\mathcal{S}_j$ to be adversarial.

provide any guarantees of state estimation for the set of misbehaving nodes $\mathcal{V} \setminus \mathcal{R}'$, since $\mathcal{V} \setminus \mathcal{R}' \subseteq \mathcal{A}$. In this case too, the subgraphs returned by Algorithm 1 have enough redundancy to ensure that Problem 3 can be solved based on our proposed approach; this fact can be established using arguments identical to those used for proving Theorem 3.5.1. In what follows, we summarize our overall approach.

3.6.1 Summary of the Resilient Distributed State Estimation Scheme

1) Each regular node $i \in \mathcal{R}$ performs the coordinate transformation $\mathbf{z}[k] = \mathbf{T}^{-1}\mathbf{x}[k]$ described in Section 3.4.1; accordingly, it identifies its detectable and undetectable eigenvalues ($\mathcal{O}_i$ and $\overline{\mathcal{O}}_i$ respectively).

2) The MEDAG construction algorithm described by Algorithm 1 is implemented for each $\lambda_j \in \Omega_U(\mathbf{A})$; graph conditions for termination of this algorithm are provided in the next section. At the end of this algorithm, each regular node i knows the subset $\mathcal{N}_i^{(j)}$ of neighbors it should use in the LFRE algorithm.

3) Each regular node i employs a locally constructed Luenberger observer (refer to Lemma 3.4.3 and the discussion preceding it) for estimating $\mathbf{z}_{\mathcal{O}_i}[k]$, namely the portion of the state $\mathbf{z}[k]$ corresponding to its detectable eigenvalues.

4) Each regular node i employs the LFRE algorithm governed by equations (3.2) and (3.3) for estimating $\mathbf{z}_{\overline{\mathcal{O}}_i}[k]$, namely the portion of the state $\mathbf{z}[k]$ corresponding to its undetectable eigenvalues.

Remark 3.6.3 *Whereas steps 1 and 2 correspond to the initial design phase of our scheme, steps 3 and 4 constitute the estimation phase. A key benefit of the proposed method is that if certain graph-theoretic conditions (to be discussed in the following section) are met, then our overall scheme provably admits a* ***fully distributed implementation*** *even under worst-case adversarial behavior.*

3.7 Feasible Graph Topologies

In this section, we characterize feasible graph topologies that guarantee the termination of the MEDAG construction algorithm described in the previous section. In other words, based on Remark 3.6.2, feasible graph topologies guarantee that Problem 3 can be solved based on our proposed approach (summarized in Section 3.6.1). We first recall the following definition from [88, 106].

Definition 3.7.1 *(r-**reachable set**) For a graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$, a set $\mathcal{S} \subset \mathcal{V}$, and an integer $r \in \mathbb{N}_+$, $\mathcal{S}$ is an r-reachable set if there exists an $i \in \mathcal{S}$ such that $|\mathcal{N}_i \setminus \mathcal{S}| \geq r$,*

Thus, if a set $\mathcal{S}$ is r-reachable, then it contains a node which has at least r neighbors outside $\mathcal{S}$. We modify the notion of a *strongly-r robust graph* from [106] as follows.

Definition 3.7.2 *(**strongly** r-**robust graph** w.r.t. $\mathcal{S}_j$) For $r \in \mathbb{N}_+$ and $\lambda_j \in \Omega_U(\mathbf{A})$, a graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ is strongly r-robust w.r.t. to the set of source nodes $\mathcal{S}_j$, if for any non-empty subset $\mathcal{C} \subseteq \mathcal{V} \setminus \mathcal{S}_j$, $\mathcal{C}$ is r-reachable.*

For an illustration of the above definitions, the reader is referred back to Figure 3.2. Figure 3.2(a) is an example of a network that is strongly 3-robust w.r.t. the set of source nodes, namely nodes $\{1, 2, 3\}$. Specifically, all subsets of $\{4, 5, 6, 7\}$ are 3-reachable (i.e., each such subset has a node that has at least 3 neighbors outside that subset).

Lemma 3.7.1 *The MEDAG construction algorithm terminates for $\lambda_j \in \Omega_U(\mathbf{A})$ if $\mathcal{G}$ is strongly $(3f + 1)$-robust w.r.t. $\mathcal{S}_j$.*

Proof We prove by contradiction. Consider any $\lambda_j \in \Omega_U(\mathbf{A})$ and let $\mathcal{G}$ be strongly $(3f + 1)$-robust w.r.t. the set of source nodes $\mathcal{S}_j$. Suppose that the MEDAG construction algorithm for λ_j does not terminate. Since the possibility of the counter $c_i(j)$ oscillating between 0 and 1 (where $i \in \mathcal{R}$) is ruled out based on our MEDAG construction algorithm, there must then exist a non-empty set $\mathcal{C} \subseteq \mathcal{V} \setminus \mathcal{S}_j$ of regular nodes that never update their counter $c_i(j)$ from 0 to 1, where $i \in \mathcal{C}$. As $\mathcal{G}$ is strongly $(3f+1)$-robust w.r.t. $\mathcal{S}_j$, it follows that $\mathcal{C}$ is $(3f+1)$-reachable, i.e., there exists a node $i \in \mathcal{C}$ which has at least $(3f+1)$ neighbors outside $\mathcal{C}$. Under the f-local adversarial model, at least $(2f+1)$ of them are regular nodes with $c_i(j) = 1$. Thus, at least $(2f+1)$ regular nodes must have transmitted the message "1" to node i. Thus, based on the rules of Algorithm 1, node i must have updated $c_i(j)$ from 0 to 1 at some point of time, leading to a contradiction. ■

Whereas the $(2f+1)$ term appears in various contexts when dealing with security problems on networks (such as distributed consensus [87, 88], broadcasting [91, 92] and optimization [89, 90]), the $(3f+1)$ term featuring in our analysis accounts for misbehavior that involves transmission of no messages by the adversarial nodes during execution of the MEDAG construction algorithm described in Section 3.6. We now present the main result of this chapter which ties together the previous results presented in this chapter, and in turn provides a connection between feasible graph topologies and the solution to Problem 3 based on our proposed approach.

Theorem 3.7.2 *Consider an LTI system* (2.1) *and a measurement model* (2.2)*. Let the communication graph* $\mathcal{G}$ *be strongly* $(3f+1)$*-robust w.r.t.* $\mathcal{S}_j, \forall \lambda_j \in \Omega_U(\mathbf{A})$*. Then, the proposed algorithm summarized in Section 3.6.1 provides a solution to Problem 3.*

Proof From Lemma 3.7.1, it follows that if $\mathcal{G}$ is strongly $(3f+1)$-robust w.r.t. $\mathcal{S}_j$ for every $\lambda_j \in \Omega_U(\mathbf{A})$, then the MEDAG construction algorithm terminates for each such eigenvalue. Combining Theorem 3.6.1, Remark 3.6.2 and Theorem 3.5.1 then leads to the desired result. ■

If the adversarial attacks are restricted to the estimation phase only, i.e., if there are no attacks during the initial MEDAG construction phase, then the following result provides a tight graph condition for our algorithm.

Theorem 3.7.3 *Consider an LTI system* (2.1) *and a measurement model* (2.2)*. Suppose adversarial behavior is restricted to the estimation phase (steps* 3 *and* 4*) of the proposed algorithm summarized in Section 3.6.1. Then, this algorithm solves Problem 3 if and only if* $\mathcal{G}$ *is strongly* $(2f+1)$*-robust w.r.t.* $\mathcal{S}_j$*,* $\forall \lambda_j \in \Omega_U(\mathbf{A})$*.*

The proof of the above result is given in Section 3.10.3. Essentially, Theorem 3.7.3 alludes to the fact that $\mathcal{G}$ contains a MEDAG $\mathcal{G}_j$ for each $\lambda_j \in \Omega_U(\mathbf{A})$ *if and only if* $\mathcal{G}$ is strongly $(2f+1)$-robust w.r.t. $\mathcal{S}_j$, $\forall \lambda_j \in \Omega_U(\mathbf{A})$. Note that although Theorem 3.7.3 provides a graph condition that is necessary and sufficient for the algorithm developed in this chapter, such a condition may not be necessary for solving Problem 3 in general.

Theorems 3.7.2 and 3.7.3 reveal that '*strong* r*-robustness w.r.t.* $\mathcal{S}_j$, $\forall \lambda_j \in \Omega_U(\mathbf{A})$' is the key topological property required for guaranteeing success of our proposed algorithm. Accordingly, given a system model (2.1) and measurement model (2.2), a network that is strongly r-robust w.r.t. $\mathcal{S}_j$, $\forall \lambda_j \in \Omega_U(\mathbf{A})$, will be called an '$r$*-feasible network*' for simplicity. We summarize certain features of an r-feasible network in the following result.

Proposition 3.7.1 *An* r*-feasible network* $\mathcal{G}$ *has the following properties.*

(i) *The graph* $\mathcal{G}' = (\mathcal{V} \cup v_{new}, \mathcal{E} \cup \mathcal{E}_{new})$*, where* v_{new} *is a new vertex added to* $\mathcal{G}$ *and* $\mathcal{E}_{new}$ *is the edge set associated with* v_{new}*, is an* r*-feasible network if* $|\mathcal{N}_{v_{new}}| \geq r$*.*

(ii) $r \leq \min_{\lambda_j \in \Omega_U(\mathbf{A})} |\mathcal{S}_j|$*.*

(iii) *Let* $\mathcal{S} = \bigcap_{\lambda_j \in \Omega_U(\mathbf{A})} \mathcal{S}_j$*. Then* $|\mathcal{N}_i| \geq r$*,* $\forall i \in \mathcal{V} \setminus \mathcal{S}$*.*

(iv) *Removal of a* k*-local set from* $\mathcal{G}$*, where* $0 < k < r$*, results in a network* $\mathcal{G}'$ *that is* $(r-k)$*-feasible.*

Proof (i) Consider any $\lambda_j \in \Omega_U(\mathbf{A})$. If v_{new} is a source node for λ_j, then it is easily seen that $\mathcal{G}'$ is strongly r-robust w.r.t. $\mathcal{S}_j$. For the case when v_{new} is not a source node for λ_j, consider any non-empty set $\mathcal{C} \subseteq \{\mathcal{V} \cup v_{new}\} \setminus \mathcal{S}_j$. If $\mathcal{C} = \{v_{new}\}$, then r-reachability of $\mathcal{C}$ follows from the fact that $|\mathcal{N}_{v_{new}}| \geq r$. In every other case, $\mathcal{C}$ contains some nodes of the original graph $\mathcal{G}$ and is hence r-reachable as $\mathcal{G}$ is r-feasible. Thus $\mathcal{G}'$ is strongly r-robust w.r.t. $\mathcal{S}_j$. A similar analysis holds for each $\lambda_j \in \Omega_U(\mathbf{A})$, leading to the desired result.

(ii) Suppose $r > |\mathcal{S}_\rho|$ where $\rho = \arg\min_{\lambda_j \in \Omega_U(\mathbf{A})} |\mathcal{S}_j|$. Since the set $\mathcal{C} = \mathcal{V} \setminus \mathcal{S}_\rho$ can be at most $|\mathcal{S}_\rho|$-reachable and $|\mathcal{S}_\rho| < r$, it follows that $\mathcal{G}$ is not r-feasible.

(iii) Suppose $i \in \mathcal{V} \setminus \mathcal{S}$ with $|\mathcal{N}_i| < r$. As $i \in \mathcal{V} \setminus \mathcal{S}$, there exists some $\lambda_j \in \Omega_U(\mathbf{A})$ such that $i \in \mathcal{V} \setminus \mathcal{S}_j$. Consider the set $\mathcal{C} = \{i\}$. As $|\mathcal{N}_i| < r$, the set $\mathcal{C}$ is not r-reachable. Thus, $\mathcal{G}$ is not strongly r-robust w.r.t. $\mathcal{S}_j$, implying that $\mathcal{G}$ is not r-feasible.

(iv) First, observe that as $\mathcal{G}$ is r-feasible and $k < r$, removal of a k-local set from $\mathcal{G}$ cannot cause the removal of an entire source node net $\mathcal{S}_j$ for any $\lambda_j \in \Omega_U(\mathbf{A})$. This follows from noting that any source set $\mathcal{S}_j$ where $\lambda_j \in \Omega_U(\mathbf{A})$ (or any set containing $\mathcal{S}_j$) will have an overlap of at least r nodes with the neighborhood of some non-source node owing to the r-feasibility of the original network. As $r > k$, such sets are not k-local. Next, pick any $\lambda_j \in \Omega_U(\mathbf{A})$ and let $\mathcal{C}$ be a non-empty subset of $\mathcal{V}' \setminus \mathcal{S}_j{}'$, where $\mathcal{V}'$ and $\mathcal{S}_j{}'$ represent the vertex set and source node set for λ_j, respectively, in $\mathcal{G}'$. Since $\mathcal{C}$ was r-reachable in $\mathcal{G}$, it contained some node v with r neighbors outside $\mathcal{C}$. While constructing $\mathcal{G}'$, node v can lose at most k of such neighbors, and hence $\mathcal{C}$ is $(r-k)$-reachable in $\mathcal{G}'$. The rest of the proof follows trivially. ■

We remark on certain implications of the above result. The first property provides a procedure for constructing r-feasible networks with N nodes (where $N > r$) starting from r-feasible networks with fewer than N nodes. The second property shows that the measurement structure of the nodes provides an upper bound on the robustness of the overall network. The third property places a lower bound on the minimum in-degree of any node that cannot estimate the entire state on its own in an r-feasible network. Finally, a direct implication of the fourth property is that a loss of k source nodes (where $k < r$) for any unstable eigenvalue of the system (possibly due to sensor failures) leaves the resulting network at least $(r-k)$-feasible if the original network is r-feasible to begin with.

3.7.1 Applicability of the Proposed Approach

Building on the insights developed in this section, we make a case for the applicability of the approach developed in this chapter by addressing the following question: How efficiently can one verify whether a given system and network is r-feasible? To answer the above question, we will exploit a connection between the 'strong r-robustness property w.r.t. a certain set of nodes' and the dynamic process of 'bootstrap percolation' on networks [107]. Given a graph $\mathcal{G}$ and a threshold $r \geq 2$, bootstrap percolation can be viewed as a process of spread of *activation* where one starts off with a set $\mathcal{I} \subseteq \mathcal{V}$ of initially active nodes. Subsequently, the process evolves over the network based on the rule that an inactive node becomes active if and only if it has at least r active neighbors, with active nodes remaining active forever. The process terminates when no more nodes become active; an initial set $\mathcal{I}$ is said to *percolate* if upon termination the final active set equals the entire node set $\mathcal{V}$. Consider the following simple, yet key observation.

Lemma 3.7.4 *Given a graph $\mathcal{G}$ and a threshold $r \geq 2$, an initial set $\mathcal{I}$ percolates via the process of bootstrap percolation if and only if $\mathcal{G}$ is strongly r-robust w.r.t. $\mathcal{I}$.*

The proof of the above result follows similar arguments as Lemma 3.7.1, and is hence omitted. Leveraging Lemma 3.7.4, we obtain the following result.

Proposition 3.7.2 *Given a system matrix $\mathbf{A} \in \mathbb{R}^{n \times n}$ (2.1), a measurement model (2.2), a communication graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ with $|\mathcal{V}| = N$, the source set $\mathcal{S}_j$ for each $\lambda_j \in sp(\mathbf{A})$, and an integer $r \geq 2$, one can verify whether the network is r-feasible in $O(nN|\mathcal{E}|)$ time.*

Proof Notice that $|\Omega_U(\mathbf{A})| \leq n$, i.e., there are at most n source sets $\mathcal{S}_j$ for which we need to verify the strong r-robustness property in Definition 3.7.2. Based on

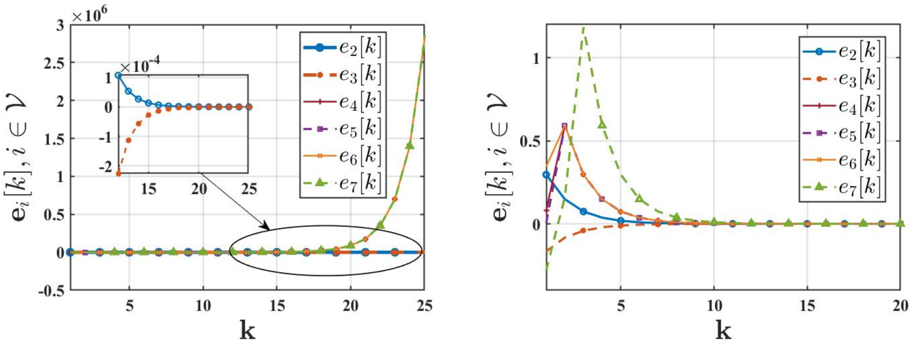
Lemma 3.7.4, for each $\mathcal{S}_j$ corresponding to some $\lambda_j \in \Omega_U(\mathbf{A})$, verifying whether $\mathcal{G}$ is strongly r-robust w.r.t. $\mathcal{S}_j$ is equivalent to verifying whether $\mathcal{S}_j$ percolates via the process of bootstrap percolation with threshold r. Thus, we analyze the complexity of simulating a bootstrap percolation process on a given network.[7] First, notice that it takes at most N iterations/rounds for a bootstrap percolation process to terminate on a network of N nodes. In each round, every inactive node checks whether it has at least r active neighbors; the entire process of checking is thus completed in $O(\sum_{i=1}^{N} d_i)$ $= O(|\mathcal{E}|)$ time, where d_i represents the in-degree of node i. Thus, for a given initial set, it takes $O(N|\mathcal{E}|)$ time to simulate the bootstrap percolation process. The result then follows readily. ■

Remark 3.7.5 *Based on the above result, one can check whether the approach developed in this chapter is applicable for a given system and network in polynomial time. This result is in stark contrast with analogous results existing in the resilient distributed consensus [87, 88] and optimization [89, 90] literature, since checking the 'robustness' condition needed for solving such problems is coNP-complete. Interestingly, leveraging the equivalence described in Lemma 3.7.4, it is possible to show that the strong r-robustness property described in Definition 3.7.2 is exhibited by various large-scale complex network models such as the Barabási-Albert (BA) preferential attachment model, the Erdős-Rényi random graph model, and the 2-dimensional random geometric graph model. A discussion on this topic can be found in Appendix 3.11.*

3.8 Simulations

Consider the system and network given by Figure 3.2. The state evolves as $x[k+1] = ax[k]$, with $a = 2$. Nodes 1, 2 and 3 are the source nodes and directly estimate the state, i.e., $y_i[k] = x[k], \forall i \in \{1, 2, 3\}$. The rest of the nodes have zero measurements. Node 1 is the only adversarial node in the network, and it simply transmits a constant

[7] Algorithm 1 essentially simulates the evolution of a bootstrap percolation process with threshold $r = (2f+1)$, provided there is no adversarial activity during the distributed implementation of such an algorithm.



(a) (b)

Fig. 3.3. Consider the system and network in Fig. 3.2. Fig. (a) depicts how a single adversary, namely node 1, can cause the estimation errors of all the non-source regular nodes (namely, nodes 4-7) to diverge when a non-resilient distributed observer is employed. Fig. (b) shows hows the proposed LFRE algorithm counteracts the effect of the adversary.

signal of magnitude $\epsilon = 0.001$ to each of its neighbors at every time-step. Each regular source node updates its state estimate based on a standard Luenberger observer as follows:

$$\hat{x}_i[k+1] = a\hat{x}_i[k] + l_i(y_i[k] - \hat{x}_i[k]), i \in \{2, 3\}, \tag{3.4}$$

with the observer gain l_i set to 1.5 (this gain is simply chosen to ensure stability). We first consider a scenario where each non-source node updates its estimate as follows: $\hat{x}_i[k+1] = a\sum_{l \in \mathcal{N}_i^{(j)}} w_{il}^{(j)} \hat{x}_l[k]$, where the weights form a convex combination, and $\mathcal{N}_i^{(j)}$ represents the neighbors of node i in the MEDAG shown in Figure 3.2(b).[8] If node 1 were to update its estimate as per (3.4), then it can be easily verified analytically that all nodes would be able to track the state asymptotically (see [58] for details). However, as seen from Figure 3.3(a), a single adversarial node (node 1 in this case) transmitting a small constant signal can cause the estimates of all the non-source nodes to diverge. This example demonstrates that although the underlying network is strongly 3-robust (i.e., has enough built-in redundancy to deal with a single

[8] For this scalar system, there is only one mode, i.e., $j = 1$.

adversarial node[9]), the non-resilient distributed observer employed above proves to be inadequate in the face of attacks. However, as seen from Figure 3.3(b), the LFRE algorithm complements the robust network structure and succeeds in counteracting the adversarial attack. For all simulations, $x[0] = 0.5$, and $\hat{x}_i[0], i \in \mathcal{V}$ is a random number between 0 and 1.

3.9 Chapter Summary

In this chapter, we studied the problem of collaboratively estimating the state of an LTI system subject to worst-case adversarial behavior. For the attack models under consideration, we identified certain necessary conditions that need to be satisfied by any system and network for the problem posed in this chapter to have a feasible solution. We then developed a local-filtering algorithm to enable each non-compromised node to recover the entire state. Finally, using a topological property called strong r-robustness, we characterized networks that guarantee success of our proposed strategy. Two notable features of our approach are as follows: (i) each step of our approach admits an attack-resilient, completely distributed implementation provided certain graph-theoretic conditions are met; and (ii) these graph-theoretic conditions can be checked in polynomial time.

3.10 Omitted Proofs

3.10.1 Proof of Theorem 3.3.3

Proof "(i)$\implies$(ii)" We prove by contraposition. Suppose statement (ii) is violated for some node $i \in \mathcal{V}$, i.e., there exists a set $\mathcal{D}_i$ such that its removal from $\mathcal{G}$ causes the pair $(\mathbf{A}, \mathbf{C}_{i\cup\mathcal{P}_i})$ to become undetectable (where $\mathcal{D}_i$ and $\mathcal{P}_i$ have the same meaning as in the statement of Theorem 3.3.3). It then follows that $\mathcal{F} = \mathcal{V} \setminus \{i \cup \mathcal{P}_i\}$ is a critical set. Suppose it is also a minimal critical set. We construct $\mathcal{G}'$ by adding directed

[9] For this example, we assume that node 1 misbehaves only during the estimation phase. Hence, Theorem 3.7.3 is applicable.

edges from a virtual node s to each node in $\mathcal{F}$.[10] Observe that $\mathcal{H} = \mathcal{D}_i$ satisfies all the properties of an f-total pair cut w.r.t. s. In particular, $\mathcal{Y} = \{i \cup \mathcal{P}_i\}$ and $\mathcal{X} = \{\mathcal{V} \setminus \{\mathcal{D}_i \cup \mathcal{Y}\}\} \cup \{s\}$. Thus, statement (i) is violated. A similar argument holds when $\mathcal{F}$ contains a minimal critical set.

"(i)$\Longleftarrow$(ii)" We again prove by contraposition. Suppose statement (i) is violated, i.e., there exists an f-total pair cut $\mathcal{H}$ w.r.t. a virtual node s corresponding to some minimal critical set $\mathcal{F}$. Consider a node i in $\mathcal{Y}$ (recall that $\mathcal{Y}$ is non-empty based on Definition 3.3.3). First consider the case when node i is not reachable from any node in $\mathcal{H}$ in the graph $\mathcal{G}$. It then follows that in the graph $\mathcal{G}$, directed paths to node i can only exist from the set $\mathcal{Y}$. But since $i \in \mathcal{Y}$ and $(\mathbf{A}, \mathbf{C}_{\mathcal{Y}})$ is not detectable, it is trivially impossible for node i to estimate the state. We thus focus on the case where node i is reachable from a certain set of nodes, say $\mathcal{D}_i$, within the set $\mathcal{H}$. Since $|\mathcal{H}| \leq 2f$ and $\mathcal{D}_i \subseteq \mathcal{H}$, we have that $|\mathcal{D}_i| \leq 2f$. It can be easily argued that the removal of $\mathcal{D}_i$ from $\mathcal{G}$ results in an induced subgraph where node i can only be reached from the set $\mathcal{Y}$. In other words, the set $\mathcal{P}_i$, as defined in the statement of Theorem 3.3.3, is a subset of $\mathcal{Y}$. As $(\mathbf{A}, \mathbf{C}_{\mathcal{Y}})$ is not detectable, it follows that $(\mathbf{A}, \mathbf{C}_{i \cup \mathcal{P}_i})$ is not detectable either, and thus statement (ii) is violated.

■

3.10.2 Proof of Theorem 3.5.1

Proof Let $\mathcal{A}$ be the (unknown) set of f-local adversaries, and consider $\mathcal{R} = \mathcal{V} \setminus \mathcal{A}$. Given a node $i \in \mathcal{R}$, the state vector $\mathbf{z}[k]$ can be partitioned into the components $\mathbf{z}_{\mathcal{O}_i}[k]$ and $\mathbf{z}_{\overline{\mathcal{O}}_i}[k]$ that correspond to the detectable and undetectable eigenvalues, respectively, of node i. Based on Lemma 3.4.3, we know that node i can estimate $\mathbf{z}_{\mathcal{O}_i}[k]$ asymptotically via a locally constructed Luenberger observer. It remains to show that node i can recover $\mathbf{z}_{\overline{\mathcal{O}}_i}[k]$, or in other words, for each $\lambda_j \in \overline{\mathcal{O}}_i$, we need to prove that $\lim_{k\to\infty} \|\hat{\mathbf{z}}_i^{(j)}[k] - \mathbf{z}^{(j)}[k]\| = 0$. To this end, consider a non-real eigenvalue $\lambda_j \in \Omega_U(\mathbf{A})$.

[10] Throughout this proof, we drop the subscript on $\mathcal{G}', \mathcal{F}$ and s, unlike the notation in Section 3.3. This is done since the subscript i is used to denote sets defined w.r.t. a node i in this proof.

As $\mathcal{G}$ contains a MEDAG for each $\lambda_j \in \Omega_U(\mathbf{A})$, the sets $\{\mathcal{L}_0^{(j)}, \mathcal{L}_1^{(j)}, \cdots, \mathcal{L}_q^{(j)}, \cdots \mathcal{L}_{T_j}^{(j)}\}$ form a partition of the set $\mathcal{R}$. We prove that each node in $\mathcal{R}$ can asymptotically estimate $\mathbf{z}^{(j)}[k]$ by inducting on the level number q.

For $q = 0$, by definition of the set $\mathcal{L}_0^{(j)}$, all nodes in $\mathcal{L}_0^{(j)}$ are regular and belong to the set $\mathcal{S}_j$, i.e., $\lambda_j \in \mathcal{O}_i$ for each node i in $\mathcal{L}_0^{(j)}$. Thus, by Lemma 3.4.3, each node in level 0 can estimate $\mathbf{z}^{(j)}[k]$ asymptotically. Notice that for any node i belonging to a level q, where $1 \leq q \leq T_j$, we have $\lambda_j \in \overline{\mathcal{O}}_i$. Consider a node i in $\mathcal{L}_1^{(j)}$ and let its error in estimation of the component $z^{(jm)}[k]$ be denoted by $e_i^{(jm)}[k] \triangleq \hat{z}_i^{(jm)}[k] - z^{(jm)}[k]$. The estimation errors of the individual components are aggregated in the vector $\mathbf{e}_i^{(j)}[k] = \hat{\mathbf{z}}_i^{(j)}[k] - \mathbf{z}^{(j)}[k]$. Subtracting $\mathbf{z}^{(j)}[k+1]$ from both sides of equation (3.3), noting that $\mathbf{z}^{(j)}[k+1] = \mathbf{W}(\lambda_j)\mathbf{z}^{(j)}[k]$ (based on the dynamics given by (3.1)), and using (3.2), we obtain

$$\mathbf{e}_i^{(j)}[k+1] = \mathbf{W}(\lambda_j) \underbrace{\begin{bmatrix} \sum_{l \in \mathcal{M}_i^{(j1)}[k]} w_{il}^{(j1)}[k] e_l^{(j1)}[k] \\ \vdots \\ \sum_{l \in \mathcal{M}_i^{(j\sigma_j)}[k]} w_{il}^{(j\sigma_j)}[k] e_l^{(j\sigma_j)}[k] \end{bmatrix}}_{\bar{\mathbf{e}}_i^{(j)}[k]}, \tag{3.5}$$

where $\sigma_j = 2a_{\mathbf{A}}(\lambda_j)$ (since λ_j is non-real). For arriving at (3.5), we used the fact that $\sum_{l \in \mathcal{M}_i^{(jm)}[k]} w_{il}^{(jm)}[k] = 1$ for every component m of $\mathbf{z}^{(j)}[k]$. We now analyze the error dynamics (3.5). To this end, for each component m of the vector $\mathbf{z}^{(j)}[k]$, we partition the set $\mathcal{N}_i^{(j)}$ into the sets $\mathcal{U}_i^{(jm)}[k]$, $\mathcal{J}_i^{(jm)}[k]$, and $\mathcal{M}_i^{(jm)}[k]$, such that the sets $\mathcal{U}_i^{(jm)}[k]$ and $\mathcal{J}_i^{(jm)}[k]$ contain f nodes each, with the highest and lowest estimate values for $z^{(jm)}[k]$ respectively, transmitted to node i at time-step k, and $\mathcal{M}_i^{(jm)}[k]$ contains the rest of the nodes in $\mathcal{N}_i^{(j)}$. According to the update rule (3.2), node i only uses estimates from the set $\mathcal{M}_i^{(jm)}[k]$ (which is non-empty at all time-steps based on the properties of a MEDAG) to compute the quantity $\bar{z}_i^{(jm)}[k]$. Now, for any component m of $\mathbf{z}^{(j)}[k]$, consider the following two cases. (i) $\mathcal{M}_i^{(jm)}[k] \cap \mathcal{A} = \emptyset$, *i.e., there are no adversarial nodes in the set* $\mathcal{M}_i^{(jm)}[k]$: in this case, all the nodes in the set $\mathcal{M}_i^{(jm)}[k]$ are regular and belong to $\mathcal{L}_0^{(j)}$ (as $\mathcal{N}_i^{(j)} \cap \mathcal{R} \subseteq \mathcal{L}_0^{(j)}$). (ii) $\mathcal{M}_i^{(jm)}[k] \cap \mathcal{A}$ *is non-empty, i.e., there are some adversarial nodes in the set* $\mathcal{M}_i^{(jm)}[k]$: based on the

f-local adversarial model, it is apparent that each of the sets $\mathcal{U}_i^{(jm)}[k]$ and $\mathcal{J}_i^{(jm)}[k]$ contain at least one regular node belonging to $\mathcal{L}_0^{(j)}$. Let u and v be two such regular nodes belonging to $\mathcal{U}_i^{(jm)}[k]$ and $\mathcal{J}_i^{(jm)}[k]$, respectively. Based on the definitions of the sets $\mathcal{U}_i^{(jm)}[k]$, $\mathcal{J}_i^{(jm)}[k]$, and $\mathcal{M}_i^{(jm)}[k]$, we have $\hat{z}_v^{(jm)}[k] \leq \hat{z}_l^{(jm)}[k] \leq \hat{z}_u^{(jm)}[k]$, and hence $e_v^{(jm)}[k] \leq e_l^{(jm)}[k] \leq e_u^{(jm)}[k]$, for every node $l \in \mathcal{M}_i^{(jm)}[k]$. In particular, since $u, v \in \mathcal{L}_0^{(j)}$, it follows that for any node $l \in \mathcal{M}_i^{(jm)}[k]$, $e_{min}^{(jm)}[k] \leq e_l^{(jm)}[k] \leq e_{max}^{(jm)}[k]$, where $e_{min}^{(jm)}[k] = \min_{u \in \mathcal{L}_0^{(j)}} e_u^{(jm)}[k]$ and $e_{max}^{(jm)}[k] = \max_{u \in \mathcal{L}_0^{(j)}} e_u^{(jm)}[k]$. This property holds for every component m of $\mathbf{z}^{(j)}[k]$. Analyzing each of the two cases, we infer that at every time-step k, each component of the vector $\bar{\mathbf{e}}_i^{(j)}[k]$ in (2.15) lies in the convex hull of the corresponding components of the error vectors $\mathbf{e}_u^{(j)}[k], u \in \mathcal{L}_0^{(j)} = \mathcal{S}_j \cap \mathcal{R}$. Based on Lemma 3.4.3, we have that $\lim_{k \to \infty} \mathbf{e}_u^{(j)}[k] = \mathbf{0}$, $\forall u \in \mathcal{S}_j \cap \mathcal{R}$, and hence it follows that $\hat{\mathbf{z}}_i^{(j)}[k]$ converges asymptotically to $\mathbf{z}^{(j)}[k]$ for every regular node i in $\mathcal{L}_1^{(j)}$.

Suppose the result holds for all levels from 0 to q (where $1 \leq q \leq T_j - 1$). It is easy to see that the result holds for all regular nodes in $\mathcal{L}_{q+1}^{(j)}$ as well, by noting the following. (i) A regular node $i \in \mathcal{L}_{q+1}^{(j)}$ has $\mathcal{N}_i^{(j)} \cap \mathcal{R} \subseteq \bigcup_{r=0}^{q} \mathcal{L}_r^{(j)}$. (ii) For each $i \in \mathcal{L}_{q+1}^{(j)}$, a similar analysis reveals that at every every time-step k, each component of the vector $\bar{\mathbf{e}}_i^{(j)}[k]$ lies in the convex hull of the corresponding components of the error vectors $\mathbf{e}_u^{(j)}[k], u \in \bigcup_{r=0}^{q} \mathcal{L}_r^{(j)}$. The desired result then follows from the induction hypothesis. An identical argument can be sketched for a real eigenvalue $\lambda_j \in \Omega_U(\mathbf{A})$, and thus the result holds for any $\lambda_j \in \Omega_U(\mathbf{A})$. We arrive at the conclusion that every node $i \in \mathcal{R}$ can asymptotically estimate $\mathbf{z}^{(j)}[k]$ for every eigenvalue $\lambda_j \in \overline{\mathcal{O}}_i$. Thus, each node $i \in \mathcal{R}$ can asymptotically estimate $\mathbf{z}[k]$, and hence $\mathbf{x}[k] = \mathbf{T}\mathbf{z}[k]$. ■

3.10.3 Proof of Theorem 3.7.3

Proof For sufficiency, it is easily noted that the conditions stated in the theorem guarantee termination of the MEDAG construction algorithm for every $\lambda_j \in \Omega_U(\mathbf{A})$. The rest of the proof for sufficiency follows identical arguments as the proof of Theorem 3.7.2.

For proving necessity, we first note that the proposed algorithm summarized in Section 3.6.1 is applicable only if the MEDAG construction algorithm (Algorithm 1) terminates for each $\lambda_j \in \Omega_U(\mathbf{A})$ and returns a subgraph $\mathcal{G}_j$ satisfying the properties of a MEDAG for all f-local sets containing $\mathcal{V} \setminus \mathcal{R}'$. Here $\mathcal{R}'$ denotes the set of nodes that behave regularly during the execution of Algorithm 1. Based on the hypothesis of the theorem, since $\mathcal{R}' = \mathcal{V}$, the existence of a MEDAG $\mathcal{G}_j \; \forall \lambda_j \in \Omega_U(\mathbf{A})$ is necessary in this case for running the LFRE algorithm. The rest of the proof proceeds via contradiction. Suppose $\mathcal{G}$ is not strongly $(2f+1)$-robust w.r.t. $\mathcal{S}_j$ for some $\lambda_j \in \Omega_U(\mathbf{A})$ and yet there exists a MEDAG $\mathcal{G}_j$ for λ_j. Since $\mathcal{G}$ is not strongly $(2f+1)$-robust w.r.t. λ_j, there exists a non-empty set $\mathcal{C} \subseteq \mathcal{V} \setminus \mathcal{S}_j$ that is not $(2f+1)$-reachable. Consider the trivial f-local set $\mathcal{A} = \emptyset$. The subgraph $\mathcal{G}_j$ must contain a partition of $\mathcal{R} = \mathcal{V} \setminus \mathcal{A} = \mathcal{V}$ into levels that satisfy the second property of a MEDAG in Definition 3.5.1. With this point in mind, let $\mathcal{C}$ be partitioned as $\mathcal{C} = \bigcup_{r=1}^{q} \mathcal{F}_r$, where $\mathcal{F}_r = \mathcal{C} \cap \mathcal{L}^{(j)}_{n_r}$ for some set of integers $\{n_1, \cdots, n_q | 1 \leq n_i \leq T_j \, \forall i \in \{1, \cdots, q\}\}$. Here, $\{\mathcal{L}^{(j)}_{n_r}\}_{r=1}^{q}$ represents a subset of the levels that partition $\mathcal{R}$ in the MEDAG $\mathcal{G}_j$ (that exists based on the hypothesis). Without loss of generality, let $n_1 < n_2 < \cdots < n_q$. Then, from the definition of a MEDAG, it follows that for any $i \in \mathcal{F}_{n_1}$, $N^{(j)}_i$ contains elements from only $\mathcal{V} \setminus \mathcal{C}$. As $\mathcal{C}$ is not $(2f+1)$-reachable, $|\mathcal{N}^{(j)}_i| < (2f+1)$, thereby violating the first property of a MEDAG in Definition 3.5.1. We thus arrive at a contradiction, and the proof is complete. ■

3.11 The Strong r-robustness Property of Random Graphs

The focus of this section is to address the following question. Given a dynamical system and an associated large-scale complex sensor network monitoring the system, under what conditions is the system and network pair r-feasible? To provide an answer to this question, we study the 'strong r-robustness' property in three relatively common random graph models for large-scale complex networks, namely the Barabási-

Albert (BA) preferential attachment model, the Erdős-Rényi random graph model, and the 2-dimensional random geometric graph model.

Consider a scenario where we are given a dynamical system and an associated wireless sensor network such that the system and network pair is r-feasible, and hence, resilient to adversarial attacks. We wish to expand the network via addition of more sensors without disrupting the r-feasibility property, i.e., we intend to tolerate the same number of adversaries as earlier. As the first property in Proposition 3.7.1 suggests, this can be achieved by continually adding new nodes with incoming edges from at least r nodes in the existing network. The specific construction where the neighbors of a new node are selected with a probability proportional to the number of edges they already have leads to the BA preferential attachment model. Such a model is thought of as a plausible mechanism for the formation of many real-world complex networks [108]. Based on our foregoing discussion, it then follows that such real-world networks would facilitate the LFRE dynamics introduced in Section 3.4.2, and would hence be resilient to the worst-case attack model considered in this chapter.

Next, we turn our attention to one of the most common mathematical models for large-scale networks, namely Erdős-Rényi random graphs [109]. We denote an Erdős-Rényi random graph on N nodes by $\mathcal{G}_{N,p}$, where all possible edges between pairs of different nodes are present independently and with the same probability p. We further note that p is in general a function of the network size N. From the perspective of a network designer, we will be interested in answering the following questions. (i) How should the size of the source sets $\mathcal{S}_j$ (for each $\lambda_j \in \Omega_U(\mathbf{A})$) scale with the size of the network to maintain r-feasibility in an Erdős-Rényi random graph? (ii) Which nodes should be chosen as the source nodes? Prior to answering these questions, we briefly remark on the notation to be used for the remainder of this section. The term w.h.p. (with high probability) will be used for events with probability tending to 1 as $N \rightarrow \infty$. Given two non-negative sequences a_N and b_N, the notation $a_N \ll b_N$

will convey the same meaning as $a_N = o(b_N)$. To make use of Lemma 3.7.4, we first recall a few definitions from [107]. Given an integer $r \geq 2$, define

$$T_c(N,p) \triangleq \left(\frac{(r-1)!}{Np^r}\right)^{\frac{1}{(r-1)}}, \; A_c(N) \triangleq \left(1 - \frac{1}{r}\right) T_c(N,p). \tag{3.6}$$

We have the following result for an Erdős-Rényi random graph model.

Proposition 3.11.1 *Given an LTI system (2.1), a measurement model (2.2), and a network modeled by an Erdős-Rényi random graph $\mathcal{G}_{N,p}$, suppose that for each $\lambda_j \in \Omega_U(\mathbf{A})$, the source set $\mathcal{S}_j(N)$ is chosen randomly.*[11] *Then, the following are true.*

(i) Let $p = p(N)$ be such that $N^{-1} \ll p \ll N^{-\frac{1}{r}}$. If for each $\lambda_j \in \Omega_U(\mathbf{A})$, $\frac{|\mathcal{S}_j(N)|}{A_c(N)} \geq 1 + \delta$, for some $\delta > 0$, and $|\mathcal{S}_j(N)| \leq \frac{N}{2}$, then $\mathcal{G}_{N,p}$ is r-feasible w.h.p. if and only if $Np - (lnN + (r-1)ln\,lnN) \rightarrow \infty$ as $N \rightarrow \infty$.

(ii) Let $p = p(N)$ be such that $p \gg N^{-\frac{1}{r}}$. If for each $\lambda_j \in \Omega_U(\mathbf{A})$, $|\mathcal{S}_j(N)| \geq r$, then $\mathcal{G}_{N,p}$ is r-feasible w.h.p.

Proof (i) If the conditions in part (i) are met, then for each $\lambda_j \in \Omega_U(\mathbf{A})$, $\mathcal{S}_j$ percolates via bootstrap percolation with threshold r on $\mathcal{G}_{N,p}$ w.h.p. based on [107, Theorem 3.2]. Lemma 3.7.4 then implies that $\mathcal{G}_{N,p}$ is strongly r-robust w.r.t. each such source set $\mathcal{S}_j$ w.h.p., i.e., $\mathcal{G}_{N,p}$ is r-feasible w.h.p. . The proof for part (ii) follows similarly by leveraging [107, Theorem 5.8] and Lemma 3.7.4. ■

Remark 3.11.1 *We glean the following insights from the above result. First, we observe that if either condition (i) or condition (ii) is met, then our proposed algorithm will enable each regular node to asymptotically estimate the state of the system w.h.p. in the presence of any $\lfloor \frac{r-1}{3} \rfloor$ locally-bounded set of Byzantine adversaries. This is a direct consequence of Theorem 3.7.2. The first part of Proposition 3.11.1 indicates that although the source sets can be chosen randomly, their size needs to scale appropriately with the size of the network to maintain r-feasibility. The second part states*

[11] By choosing $\mathcal{S}_j(N)$ randomly, we imply that the measurement set needed to detect λ_j is allocated to $|\mathcal{S}_j(N)|$ nodes picked uniformly at random. The notation $\mathcal{S}_j(N)$ is used to explicitly point out that the size of the source sets scales with the size of the network.

that if the probability of edge formation is large enough, then it suffices to pick source sets of constant size equal to the bare minimum required for achieving r-feasibility (which equals r based on part (ii) of Proposition 3.7.1).

Among the three random graph models mentioned earlier, the one most relevant to our cause is the two-dimensional random geometric graph (RGG) model [110]. RGGs are typically used to model networks where a notion of spatial proximity governs the interaction between the nodes. A wireless sensor network where randomly deployed nodes communicate with nodes only in a geographical vicinity, constitutes an ideal setup for an RGG model [111]. We will consider a two-dimensional RGG model generated by first placing N nodes randomly within the unit square $[0,1]^2$. Undirected edges are placed between two nodes if and only if the Euclidean distance between such nodes is at most $d(N)$, where $d(N)$ is a positive number that may depend on the network size N. We will denote such a RGG by $\mathcal{G}_{N,d(N)}$.

Like the Erdős-Rényi case, our focus will be on understanding how the source sets should be chosen to ensure r-feasibility of $\mathcal{G}_{N,d(N)}$ with high probability. To provide such a characterization, we first recall a few functions from [110]. Let $H(x) \triangleq x\,lnx - x + 1$ be defined on $[0,\infty)$ and $J(x) \triangleq lnx - 1 + x^{-1}$ be defined on $(0,\infty)$. Furthermore, let $J_R^{-1} : [0,\infty) \rightarrow [1,\infty)$ denote the inverse of $J(x)$ when the domain of $J(x)$ is $[1,\infty)$. We then have the following result.

Proposition 3.11.2 *Given an LTI system (2.1) and a measurement model (2.2), let the communication graph be modeled by the RGG $\mathcal{G}_{N,d(N)}$, where $d(N) = \sqrt{\frac{a\,lnN}{\pi N}}$ and $a > 1$. For each $\lambda_j \in \Omega_U(\mathbf{A})$, let a node be chosen as a source node for λ_j with a probability p independently of the other nodes in the network. Let $r = \gamma a\,lnN$, where $\gamma \in (0, \frac{1}{5\pi})$. Suppose $a \geq \frac{5\pi}{H(5\pi\gamma)}$ and*

$$p \geq \min\left\{\gamma, \frac{5\pi\gamma}{J_R^{-1}(\frac{1}{a\gamma})}\right\}. \tag{3.7}$$

Then, $\mathcal{G}_{N,d(N)}$ is r-feasible.[12]

[12] A RGG $\mathcal{G}_{N,d(N)}$ is connected w.h.p. for $d(N) > \sqrt{\frac{lnN}{\pi N}}$. The choice of $a > 1$ thus allows one to deal with an asymptotically connected $\mathcal{G}_{N,d(N)}$.

Proof For each $\lambda_j \in \Omega_U(\mathbf{A})$, if the source set $\mathcal{S}_j$ is chosen as described above, then it percolates $\mathcal{G}_{N,d(N)}$ w.h.p. if the conditions of the proposition are met [110, Theorem 4]. The result then follows from Lemma 3.7.4. ■

Remark 3.11.2 *In an attack-prone wireless sensor network, one might be interested in tolerating f-local adversarial sets where the paramater f scales with the size of the network. Such a possibility is captured by Proposition 3.11.2, based on which, $\lfloor \frac{\gamma a\, ln N - 1}{3} \rfloor$-local Byzantine adversarial sets can be accounted for by our algorithm.*

4. DISTRIBUTED STATE ESTIMATION OVER TIME-VARYING GRAPHS: EXPLOITING THE AGE-OF-INFORMATION

In this chapter, we study the problem of designing a distributed observer for an LTI system over a time-varying communication graph. Existing approaches to this problem either (i) make restrictive assumptions on the dynamical model; or (ii) make restrictive assumptions on the sequence of communication graphs; or (iii) require multiple consensus iterations between consecutive time-steps of the dynamics; or (iv) require higher-dimensional observers. In contrast, we develop a distributed observer that operates on a single time-scale, is of the same dimension as that of the state, and works under mild assumptions. Specifically, our communication model only requires strong-connectivity to be preserved over non-overlapping, contiguous intervals that are even allowed to grow unbounded over time. We show that under suitable conditions that bound the growth of such intervals, joint observability is sufficient to track the state of any discrete-time LTI system exponentially fast, at any desired rate. We also show that under a suitable selection of the observer gains, one can achieve finite-time convergence. The key to our approach is the notion of a "freshness-index" that keeps track of the age-of-information being diffused across the network. Such indices enable nodes to reject stale estimates of the state, and, in turn, contribute to stability of the error dynamics. Our proof of convergence is self-contained, and employs simple arguments from linear system theory and graph theory.

4.1 Introduction

Given a discrete-time LTI system $\mathbf{x}[k+1] = \mathbf{A}\mathbf{x}[k]$, and a linear measurement model $\mathbf{y}[k] = \mathbf{C}\mathbf{x}[k]$, a classical result in control theory states that one can design an observer that generates an asymptotically correct estimate $\hat{\mathbf{x}}[k]$ of the state $\mathbf{x}[k]$, if and only if the pair $(\mathbf{A}, \mathbf{C})$ is detectable [112]. Additionally, if the pair $(\mathbf{A}, \mathbf{C})$ is observable, then one can achieve exponential convergence at any desired convergence rate. Over the last couple of decades, significant effort has been directed towards studying the distributed counterpart of the above problem, wherein observations of the process are distributed among a set of sensors modeled as nodes of a communication graph [1, 17, 26, 27, 30–32, 58, 102, 113–115]. A fundamental question that arises in this context is as follows: What are the minimal requirements on the measurement structure of the nodes and the underlying communication graph that guarantee the existence of a distributed observer?[1] The question posed above was answered only recently in [1, 30, 58, 102, 113, 114] for static graphs. However, when the underlying network changes with time, very little is known about the distributed state estimation problem - a gap that we intend to bridge in this chapter. Our study is motivated by applications in environmental monitoring tasks using mobile robot teams [4, 5], where time-varying communication graphs arise naturally either as a consequence of robot-mobility, or due to packet drops and link failures typical in wireless sensor networks [116].

4.1.1 Related Work

The existing literature on the design of distributed observers can be broadly classified in terms of the assumptions made on the system model, the observation model, and the network structure. Recent works on this topic [1, 30, 58, 102, 113, 114] that make minimal system- and graph-theoretic assumptions can be further classified based on a finer set of attributes, as follows. (i) Does the approach require multiple consensus iterations between two consecutive time-steps of the dynamics?[2] (ii) What is the dimension of the estimator maintained by each node? (iii) Is the convergence

[1] Recall from Chapter 2 that by a distributed observer, we imply a set of state estimate update and information exchange rules that enable each node to track the entire state asymptotically.

[2] Such approaches, referred to as two-time-scale approaches, incur high communication cost, and might hence be prohibitive for real-time applications.

asymptotic, or in finite-time? (iv) In case the convergence is asymptotic, can the convergence rate be controlled? The techniques proposed in [1, 30, 58, 102, 113, 114] operate on a single-time-scale, and those in [58, 102, 113, 114] require observers of dimension no more than that of the state of the system. The notion of convergence in each of the papers [1, 30, 58, 102, 113, 114] is asymptotic; the ones in [30, 102, 114] can achieve exponential convergence at *any* desired rate. For dynamic networks, there is much less work. Two recent papers [17] and [117] provide theoretical guarantees for certain specific classes of time-varying graphs; while the former proposes a two-time-scale approach, the latter develops a single-time-scale algorithm. However, the extent to which such results can be generalized has remained an open question.

Contributions: The main contribution of this chapter is the development of a single-time-scale distributed state estimation algorithm that requires each node to maintain an estimator of dimension equal to that of the state (along with some simple counters), and works under assumptions that are remarkably mild in comparison with those in the existing literature. Specifically, in terms of the observation model, we only require joint observability, i.e., the state is observable w.r.t. the collective observations of the nodes. This assumption is quite standard, and can be relaxed to joint detectability, with appropriate implications for the convergence rate.

However, the key departure from existing literature comes from the generality of our communication model, introduced formally in Section 4.2. Based on this model, we require strong-connectivity to be preserved over non-overlapping, contiguous intervals that can even grow linearly with time at a certain rate. In other words, we allow the inter-communication intervals between the nodes to potentially grow unbounded. Even under the regime of such sparse communications, we establish that our proposed approach can track the state of any discrete-time LTI system exponentially fast, at *any* desired convergence rate. While all the works on distributed state estimation that we are aware of provide an asymptotic analysis, estimating the state of the system in *finite-time* might be crucial in certain safety-critical applications. To this end, we show that under a suitable selection of the observer gains, one can achieve finite-time convergence based on our approach. To put our results into context, it is instructive to compare them with the work closest to ours, namely [17]. In [17], the authors study a continuous-time analog of the problem under consideration, and develop a solution that leverages an elegant connection to the problem of distributed linear-equation solving [118]. In contrast to our technique, the one in [17] is inherently a two-time-scale approach, requires each node to maintain and update auxiliary state estimates, and works under the assumption that the communication graph is strongly-connected at every instant.

Our work is also related to the vast literature on consensus [119] and distributed optimization [120] over time-varying graphs, where one assumes strong-connectivity to be typically preserved over uniformly bounded intervals - a special case of our communication model. It is important to recognize that, in contrast to these settings, the problem at hand requires tracking potentially unstable external dynamics, making the stability analysis considerably more challenging. In particular, one can no longer directly leverage convergence properties of products of stochastic matrices. From a system-theoretic point of view, the error dynamics under our communication model evolves based on a switched linear system, where certain modes are potentially unstable. Thus, one way to analyze such dynamics is by drawing on ideas from the switched system stability literature [121]. However, such an analysis can potentially obscure the role played by the network structure. Instead, we directly exploit the interplay between the structure of the changing graph patterns on the one hand, and the evolution of the estimation error dynamics on the other. Doing so, we provide a comprehensive stability analysis of our estimation algorithm employing simple, self-contained arguments from linear system theory and graph theory.

The key idea behind our approach is the use of a suitably defined "freshness-index" that keeps track of the age-of-information being diffused across the network. Loosely speaking, the "freshness-index" of a node quantifies the extent to which its estimate of the state is outdated. Exchanging such indices enables a node to assess, in real-time, the quality of the estimate of a neighbor. Accordingly, it can reject estimates that are relatively stale - a fact that contributes to the stability of the error dynamics. We point out that while this is perhaps the first use of the notion of age-of-information (AoI) in a networked control/estimation setting, such a concept has been widely employed in the study of various queuing-theoretic problems arising in wireless networks [122–124].[3]

To sum up, in this chapter, we propose the first single-time-scale distributed state estimation algorithm that provides both finite-time and exponentially fast convergence guarantees, under significantly milder assumptions on the time-varying graph sequences than those in the existing literature.

The results in this chapter appeared in a preliminary form as [125]. These results were then significantly generalized in the pre-print [126].

4.2 Problem Formulation and Background

System and Measurement Model: As in Chapter 2, we are interested in collaborative state estimation of a discrete-time LTI system of the form:

$$\mathbf{x}[k+1] = \mathbf{A}\mathbf{x}[k], \tag{4.1}$$

where $k \in \mathbb{N}$ is the discrete-time index, $\mathbf{A} \in \mathbb{R}^{n \times n}$ is the system matrix, and $\mathbf{x}[k] \in \mathbb{R}^n$ is the state of the system.[4] A network of sensors, modeled as nodes of a communication graph, obtain partial measurements of the state of the above process as follows:

$$\mathbf{y}_i[k] = \mathbf{C}_i\mathbf{x}[k], \tag{4.2}$$

where $\mathbf{y}_i[k] \in \mathbb{R}^{r_i}$ represents the measurement vector of the i-th node at time-step k, and $\mathbf{C}_i \in \mathbb{R}^{r_i \times n}$ represents the corresponding observation matrix. Let $\mathbf{y}[k] = \begin{bmatrix}\mathbf{y}_1^T[k] & \cdots & \mathbf{y}_N^T[k]\end{bmatrix}^T$ and $\mathbf{C} = \begin{bmatrix}\mathbf{C}_1^T & \cdots & \mathbf{C}_N^T\end{bmatrix}^T$ represent the collective measurement

[3] The notion of age-of-information (AoI) was first introduced in [122] as a performance metric to keep track of real-time status updates in a communication system. In a wireless network, it measures the time elapsed since the generation of the packet most recently delivered to the destination. In Section 4.4, we will see how such a concept applies to the present setting.

[4] Recall that we use $\mathbb{N}$ and $\mathbb{N}_+$ to denote the set of non-negative integers and the set of positive integers, respectively.

vector at time-step k, and the collective observation matrix, respectively. The goal of each node i in the network is to generate an asymptotically correct estimate $\hat{\mathbf{x}}_i[k]$ of the true dynamics $\mathbf{x}[k]$. It may not be possible for any node i in the network to accomplish this task in isolation, since the pair $(\mathbf{A}, \mathbf{C}_i)$ may not be detectable in general. Throughout the chapter, we will only assume that the pair $(\mathbf{A}, \mathbf{C})$ is observable; the subsequent developments can be readily generalized to the case when $(\mathbf{A}, \mathbf{C})$ is detectable.

Communication Network Model: As is evident from the above discussion, information exchange among nodes is necessary for all nodes to estimate the full state. At each time-step $k \in \mathbb{N}$, the available communication channels are modeled by a directed communication graph $\mathcal{G}[k] = (\mathcal{V}, \mathcal{E}[k])$, where $\mathcal{V} = \{1, \ldots, N\}$ represents the set of nodes, and $\mathcal{E}[k]$ represents the edge set of $\mathcal{G}[k]$ at time-step k. Specifically, if $(i, j) \in \mathcal{E}[k]$, then node i can send information directly to node j at time-step k; in such a case, node i will be called a neighbor of node j at time-step k. We will use $\mathcal{N}_i[k]$ to represent the set of all neighbors (excluding node i) of node i at time-step k.

In Chapter 2, we studied the special case when $\mathcal{G}[k] = \mathcal{G}\ \forall k \in \mathbb{N}$, i.e., when $\mathcal{G}$ is a static, directed communication graph. For this setting, it was shown that the necessary and sufficient condition (on the system and network) to solve the distributed state estimation problem is the joint detectability of each source component of $\mathcal{G}$.[5] The subject of this chapter is to extend the above result to scenarios where the underlying communication graph is allowed to change over time. To this end, let the union graph over an interval $[k_1, k_2], 0 \leq k_1 \leq k_2$, denoted $\bigcup\limits_{\tau=k_1}^{k_2} \mathcal{G}[\tau]$, indicate a graph with vertex set equal to $\mathcal{V}$, and edge set equal to the union of the edge sets of the individual graphs appearing over the interval $[k_1, k_2]$. Based on this convention, we now formally describe the communication patterns (induced by the sequence $\{G[k]\}_{k=0}^{\infty}$) that are considered in this paper. We assume that there exists a sequence $\mathbb{I} = \{t_0, t_1, \ldots\}$ of increasing time-steps with $t_0 = 0$ and each $t_i \in \mathbb{N}$, satisfying the following conditions.

[5] Recall that a source component of a static, directed graph is a strongly connected component with no incoming edges.

(C1) Define the mapping $f : \mathbb{I} \to \mathbb{N}_+$ as $f(t_q) = t_{q+1} - t_q, \forall t_q \in \mathbb{I}$. We assume that $f(t_q)$ is a non-decreasing function of its argument.

(C2) For each $k \in \mathbb{N}$, let $m(k) \triangleq \max\{t_q \in \mathbb{I} : t_q \leq k\}$, and $M(k) \triangleq \min\{t_q \in \mathbb{I} : t_q > k\}$. Define $g : \mathbb{N} \to \mathbb{N}_+$ as $g(k) = M(k) - m(k) = f(m(k))$. Then, we assume that the following holds:

$$\limsup_{k\to\infty} \frac{2(N-1)g(k)}{k} = \delta < 1. \tag{4.3}$$

(C3) For each $t_q \in \mathbb{I}$, we assume that $\bigcup_{\tau=t_q}^{t_{q+1}-1} \mathcal{G}[\tau]$ is strongly-connected.

Let us discuss what the above conditions mean. Condition (C1) in conjunction with condition (C3) tells us that the intervals over which strong-connectivity is preserved are non-decreasing in length (evident from the monotonicity of the function $f(\cdot)$).[6] Essentially, our aim here is to come up with distributed estimators that function correctly despite sparse communication; hence, we allow for potentially growing inter-communication intervals. Since we place no assumptions at all on the spectrum of the $\mathbf{A}$ matrix, stability of the estimation error process imposes natural restrictions on how fast the inter-communication intervals can be allowed to grow. Condition (C2) formalizes this intuition by constraining such intervals to grow at most linearly at a certain rate. Notably, conditions (C1)-(C3) cover a very general class of time-varying graph sequences. In particular, we are unaware of any other work that allows the inter-communication intervals to grow unbounded for the problem under consideration.

Consider the following two examples. (i) The mapping f satisfies $f(t_q) = T, \forall t_q \in \mathbb{I}$, where T is some positive integer. (ii) The mapping f satisfies $f(t_q) = \lfloor \sqrt{t_q + 1} \rfloor, \forall t_q \in \mathbb{I}$. It is easy to verify that (C2) is satisfied in each case. While example (i) represents the standard "joint strong-connectivity" setting where the inter-communication intervals remain bounded, example (ii) deviates from existing literature by allowing the inter-communication intervals to grow unbounded.

[6] Our results can be generalized to account for the case when $f(\cdot)$ is non-monotonic by suitably modifying condition (C2).

Background: For communication graphs satisfying conditions (C1)-(C3) as described above, our **objective** will be to design a distributed algorithm that ensures $\lim_{k\to\infty} \|\hat{\mathbf{x}}_i[k] - \mathbf{x}[k]\| = 0, \forall i \in \mathcal{V}$, with $\hat{\mathbf{x}}_i[k]$ representing the estimate of the state $\mathbf{x}[k]$ maintained by node $i \in \mathcal{V}$. To this end, we recall the following result from Chapter 2.

Lemma 4.2.1 *Given a system matrix* $\mathbf{A}$*, and a set of* N *sensor observation matrices* $\mathbf{C}_1, \mathbf{C}_2, \ldots, \mathbf{C}_N$*, define* $\mathbf{C} \triangleq \begin{bmatrix}\mathbf{C}_1^T & \cdots & \mathbf{C}_N^T\end{bmatrix}^T$. *Suppose* $(\mathbf{A}, \mathbf{C})$ *is observable. Then, there exists a similarity transformation matrix* $\mathbf{T}$ *that transforms the pair* $(\mathbf{A}, \mathbf{C})$ *to* $(\bar{\mathbf{A}}, \bar{\mathbf{C}})$*, such that*

$$\bar{\mathbf{A}} = \left[\begin{array}{c|cc|cc} \mathbf{A}_{11} & \multicolumn{4}{c}{\mathbf{0}} \\ \hline \mathbf{A}_{21} & \mathbf{A}_{22} & & \multicolumn{2}{c}{\mathbf{0}} \\ \cline{2-5} \vdots & \vdots & & \ddots & \vdots \\ \mathbf{A}_{N1} & \mathbf{A}_{N2} & \cdots & \mathbf{A}_{N(N-1)} & \multicolumn{1}{|c}{\mathbf{A}_{NN}} \end{array}\right], \tag{4.4}$$

$$\bar{\mathbf{C}} = \begin{bmatrix} \bar{\mathbf{C}}_1 \\ \bar{\mathbf{C}}_2 \\ \vdots \\ \bar{\mathbf{C}}_N \end{bmatrix} = \left[\begin{array}{c|c|cc} \mathbf{C}_{11} & \multicolumn{3}{c}{\mathbf{0}} \\ \hline \mathbf{C}_{21} & \mathbf{C}_{22} & \multicolumn{2}{c}{\mathbf{0}} \\ \hline \vdots & \vdots & \vdots & \vdots \\ \mathbf{C}_{N1} & \mathbf{C}_{N2} & \cdots \mathbf{C}_{N(N-1)} & \mathbf{C}_{NN} \end{array}\right],$$

and the pair $(\mathbf{A}_{ii}, \mathbf{C}_{ii})$ *is observable* $\forall i \in \{1, 2, \ldots, N\}$.

We use the matrix $\mathbf{T}$ given by the above lemma to perform the coordinate transformation $\mathbf{z}[k] = \mathbf{T}^{-1}\mathbf{x}[k]$, yielding:

$$\begin{aligned} \mathbf{z}[k+1] &= \bar{\mathbf{A}}\mathbf{z}[k], \\ \mathbf{y}_i[k] &= \bar{\mathbf{C}}_i\mathbf{z}[k], \quad \forall i \in \{1, \ldots, N\}, \end{aligned} \tag{4.5}$$

where $\bar{\mathbf{A}} = \mathbf{T}^{-1}\mathbf{A}\mathbf{T}$ and $\bar{\mathbf{C}}_i = \mathbf{C}_i\mathbf{T}$ are given by (4.4). Commensurate with the structure of $\bar{\mathbf{A}}$, the vector $\mathbf{z}[k]$ is of the following form:

$$\mathbf{z}[k] = \begin{bmatrix}\mathbf{z}^{(1)}[k]^T & \cdots & \mathbf{z}^{(N)}[k]^T\end{bmatrix}^T, \tag{4.6}$$

where $\mathbf{z}^{(j)}[k]$ will be referred to as the j-th sub-state. By construction, since the pair $(\mathbf{A}_{jj}, \mathbf{C}_{jj})$ is locally observable w.r.t. the measurements of node j, node j will be viewed as the unique source node for sub-state j. In this sense, the role of node j will be to ensure that each non-source node $i \in \mathcal{V} \setminus \{j\}$ maintains an asymptotically correct estimate of sub-state j. For a time-invariant strongly-connected graph, this was achieved in Chapter 2 by first constructing a spanning tree rooted at node j, and then requiring nodes to only listen to their parents in such a tree for estimating sub-state j. We showed that the resulting unidirectional flow of information from the source j to the rest of the network guarantees stability of the error process for sub-state j.

However, the above strategy is no longer applicable when the underlying communication graph is time-varying, for the following reasons. (i) For a given sub-state j, there may not exist a common spanning tree rooted at node j in each graph $\mathcal{G}[k], k \in \mathbb{N}$. (ii) Assuming that a specific spanning tree rooted at node j is guaranteed to repeat at various points in time (not necessarily periodically), is restrictive, and qualifies as only a special case of conditions (C1)-(C3). (iii) Suppose for simplicity that $\mathcal{G}[k]$ is strongly-connected at each time-step (as in [17]), and hence, there exists a spanning tree $\mathcal{T}_j[k]$ rooted at node j in each such graph. For estimating sub-state j, suppose consensus at time-step k is performed along the spanning tree $\mathcal{T}_j[k]$. As we demonstrate in the next section, switching between such spanning trees can lead to unstable error processes over time. Thus, if one makes no further assumptions on the system model beyond joint observability, or on the sequence of communication graphs beyond conditions (C1)-(C3), ensuring stability of the estimation error dynamics becomes a challenging proposition. Nonetheless, we develop a simple algorithm in Section 4.4 that bypasses the above problems. In the next section, we provide an example that helps to build the intuition behind this algorithm.

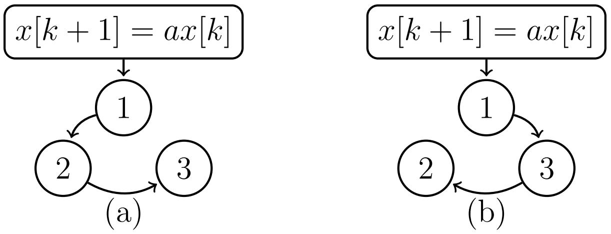


Fig. 4.1. An LTI system is monitored by a network of 3 nodes, where the communication graph $\mathcal{G}[k]$ switches between the two graphs shown above.

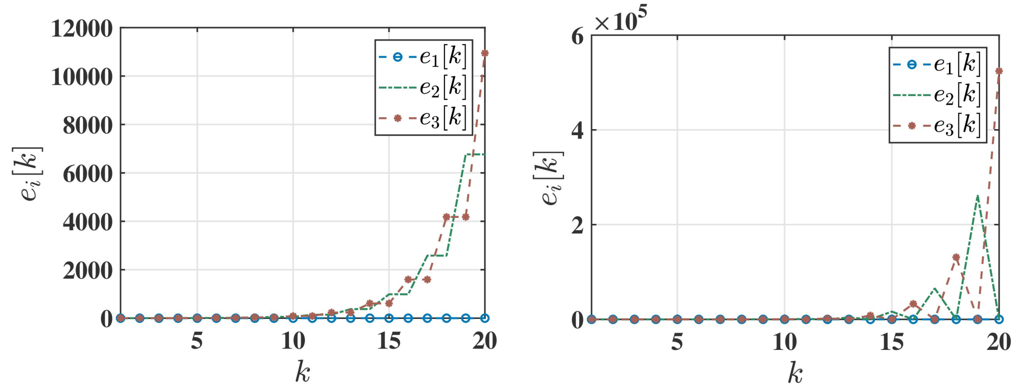


Fig. 4.2. Estimation error plots of the nodes for the model in Figure 4.1. Simulations are performed for a model where $a = 2$. The figure on the left corresponds to the case where consensus weights are distributed uniformly among neighbors, while the one on the right is the case where weights are placed along a tree rooted at node 1.

4.3 An Illustrative Example

Consider a network of 3 nodes monitoring a scalar unstable process $x[k+1] = ax[k]$, as shown in Figure 4.1. The communication graph $\mathcal{G}[k]$ switches between the two topologies shown in Figure 4.1. Specifically, $\mathcal{G}[k]$ is the graph in Figure 4.1(a) at all even time-steps, and the one in 4.1(b) at all odd time-steps. Node 1 is the only node with non-zero measurements, and thus acts as the source node for this network.

Suppose for simplicity that it perfectly measures the state at all time-steps, i.e., its state estimate is $\hat{x}_1[k] = x[k], \forall k \in \mathbb{N}$. Given this setup, a standard consensus based state estimate update rule would take the form (see for example [1, 17, 58]):

$$\hat{x}_i[k+1] = a\left(\sum_{j\in\mathcal{N}_i[k]\cup\{i\}} w_{ij}[k]\hat{x}_j[k]\right), i \in \{2,3\}, \tag{4.7}$$

where the weights $w_{ij}[k]$ are non-negative, and satisfy $\sum_{j\in\mathcal{N}_i[k]\cup\{i\}} w_{ij}[k] = 1, \forall k \in \mathbb{N}$. The key question is: *How should the consensus weights be chosen to guarantee stability of the estimation errors of nodes 2 and 3?* Even for this simple example, if such weights are chosen naively, then the errors may grow unbounded over time. To see this, consider the following two choices: (1) consensus weights are distributed evenly over the set $\mathcal{N}_i[k] \cup \{i\}$, and (2) consensus weights are placed along the tree rooted at node 1. In each case, the error dynamics are unstable, as depicted in Figure 4.2. To overcome this problem, suppose nodes 2 and 3 are aware of the fact that node 1 has perfect information of the state. Since nodes 2 and 3 have no measurements of their own, intuitively, it makes sense that they should place their consensus weights entirely on node 1 whenever possible. The trickier question for node 2 (resp., node 3) is to decide *when* it should listen to node 3 (resp., node 2). Let us consider the situation from the perspective of node 2. At time-step 0, it adopts the information of node 1, and hence, the error of node 2 is zero at time-step 1. However, the error of node 3 is not necessarily zero at time-step 1. Consequently, if node 2 places a non-zero consensus weight on the estimate of node 3 at time-step 1, its error at time-step 2 might assume a non-zero value. Clearly, at time-step 1, node 2 is better off rejecting the information from node 3, and simply running open-loop. *The main take-away point here is that adoption or rejection of information from a neighbor should be based on the quality of information that such a neighbor has to offer.* In particular, a node that has come in contact with node 1 more recently is expected to have better information about the state than the other. Thus, to dynamically evaluate the quality of an estimate, the above reasoning suggests the need to introduce a metric that keeps track of how recent that estimate is with respect to (w.r.t.) the estimate of the source node 1. In the following section, we formalize this idea by introducing such a metric.

4.4 Algorithm

Building on the intuition developed in the previous section, we introduce a new approach to designing distributed observers for a general class of time-varying networks. The main idea is the use of a "freshness-index" that keeps track of how delayed the estimates of a node are w.r.t. the estimates of a source node. Specifically, for updating its estimate of $\mathbf{z}^{(j)}[k]$, each node $i \in \mathcal{V}$ maintains and updates at every time-step a freshness-index $\tau_i^{(j)}[k]$. At each time-step $k \in \mathbb{N}$, the index $\tau_i^{(j)}[k]$ plays the following role: it determines whether node i should adopt the information received from one of its neighbors in $\mathcal{N}_i[k]$, or run open-loop, for updating $\hat{\mathbf{z}}_i^{(j)}[k]$, where $\hat{\mathbf{z}}_i^{(j)}[k]$ represents the estimate of $\mathbf{z}^{(j)}[k]$ maintained by node i. In case it is the former, it also indicates which specific neighbor in $\mathcal{N}_i[k]$ node i should listen to at time-step k; this piece of information is particularly important for the problem under consideration, and ensures stability of the error process. The rules that govern the updates of the freshness indices $\tau_i^{(j)}[k]$, and the estimates of the j-th sub-state $\mathbf{z}^{(j)}[k]$, are formally stated in Algorithm 2. In what follows, we describe each of these rules.

Discussion of Algorithm 2: Consider any sub-state $j \in \{1, \ldots, N\}$. Each node $i \in \mathcal{V}$ maintains an index $\tau_i^{(j)}[k] \in \{\omega\} \cup \mathbb{N}$, where ω is a dummy value. Specifically, $\tau_i^{(j)}[k] = \omega$ represents an "infinite-delay" w.r.t. the estimate of the source node for sub-state j, namely node j, i.e., it represents that node i has not received any information (either directly or indirectly) from node j regarding sub-state j up to time-step k. For estimation of sub-state j, since delays are measured w.r.t. the source node j, node j maintains its freshness-index $\tau_j^{(j)}[k]$ at zero for all time, to indicate a zero delay w.r.t. itself. For updating its estimate of $\mathbf{z}^{(j)}[k]$, it uses only its own information, as is evident from Eq. (4.8).

Algorithm 2

```
1: Initialization: $\tau_j^{(j)}[0] = 0, \tau_i^{(j)}[0] = \omega, \forall i \in \mathcal{V} \setminus \{j\}$.
2: Update Rules for the Source Node: Node $j$ maintains $\tau_j^{(j)}[k] = 0, \forall k \in \mathbb{N}$. It updates $\hat{\mathbf{z}}_j^{(j)}[k]$ as:
       $$\hat{\mathbf{z}}_j^{(j)}[k+1] = \mathbf{F}_{jj}\hat{\mathbf{z}}_j^{(j)}[k] + \sum_{q=1}^{(j-1)} \mathbf{G}_{jq}\hat{\mathbf{z}}_j^{(q)}[k] + \mathbf{L}_j\mathbf{y}_j[k], \qquad (4.8)$$
   where $\mathbf{F}_{jj} = (\mathbf{A}_{jj} - \mathbf{L}_j\mathbf{C}_{jj})$, $\mathbf{G}_{jq} = (\mathbf{A}_{jq} - \mathbf{L}_j\mathbf{C}_{jq})$, and $\mathbf{L}_j$ is an observer gain to be designed later.
3: Update Rules for the Non-Source Nodes: Each non-source node $i \in \mathcal{V} \setminus \{j\}$ operates as follows.
4: Case 1: $\tau_i^{(j)}[k] = \omega$. Define $\mathcal{M}_i^{(j)}[k] \triangleq \{l \in \mathcal{N}_i[k] : \tau_l^{(j)}[k] \neq \omega\}$. If $\mathcal{M}_i^{(j)}[k] \neq \emptyset$, let $u \in \text{argmin}_{l \in \mathcal{M}_i^{(j)}[k]} \tau_l^{(j)}[k]$. Node $i$ updates $\tau_i^{(j)}[k]$ and $\hat{\mathbf{z}}_i^{(j)}[k]$ as:
       $$\tau_i^{(j)}[k+1] = \tau_u^{(j)}[k] + 1, \qquad (4.9)$$
       $$\hat{\mathbf{z}}_i^{(j)}[k+1] = \mathbf{A}_{jj}\hat{\mathbf{z}}_u^{(j)}[k] + \sum_{q=1}^{(j-1)} \mathbf{A}_{jq}\hat{\mathbf{z}}_i^{(q)}[k]. \qquad (4.10)$$
   If $\mathcal{M}_i^{(j)}[k] = \emptyset$, then
       $$\tau_i^{(j)}[k+1] = \omega, \qquad (4.11)$$
       $$\hat{\mathbf{z}}_i^{(j)}[k+1] = \mathbf{A}_{jj}\hat{\mathbf{z}}_i^{(j)}[k] + \sum_{q=1}^{(j-1)} \mathbf{A}_{jq}\hat{\mathbf{z}}_i^{(q)}[k]. \qquad (4.12)$$
5: Case 2: $\tau_i^{(j)}[k] \neq \omega$. Define $\mathcal{F}_i^{(j)}[k] \triangleq \{l \in \mathcal{M}_i^{(j)}[k] : \tau_l^{(j)}[k] < \tau_i^{(j)}[k]\}$, where $\mathcal{M}_i^{(j)}[k]$ is as defined in line 4. If $\mathcal{F}_i^{(j)}[k] \neq \emptyset$, let $u \in \text{argmin}_{l \in \mathcal{F}_i^{(j)}[k]} \tau_l^{(j)}[k]$. Node $i$ then updates $\tau_i^{(j)}[k]$ as per (4.9), and $\hat{\mathbf{z}}_i^{(j)}[k]$ as per (4.10). If $\mathcal{F}_i^{(j)}[k] = \emptyset$, then $\tau_i^{(j)}[k]$ is updated as
       $$\tau_i^{(j)}[k+1] = \tau_i^{(j)}[k] + 1, \qquad (4.13)$$
   and $\hat{\mathbf{z}}_i^{(j)}[k]$ is updated as per (4.12).
```

Every other node starts out with an "infinite-delay" ω w.r.t. the source (line 1 of Algo. 2). The freshness-index of a node $i \in \mathcal{V} \setminus \{j\}$ changes from ω to a finite value when it comes in contact with a neighbor with a finite delay, i.e., with a freshness-index that is not ω (line 4 of Algo. 2). At this point, we say that $\tau_i^{(j)}[k]$ has been "triggered". Once triggered, at each time-step k, a non-source node i will adopt the information of a neighbor $l \in \mathcal{N}_i[k]$ only if node l's estimate of $\mathbf{z}^{(j)}[k]$ is "more fresh" relative to its own, i.e., only if $\tau_l^{(j)}[k] < \tau_i^{(j)}[k]$.[7] Among the set of neighbors in $\mathcal{M}_i^{(j)}[k]$ (if $\tau_i^{(j)}[k]$ has not yet been triggered), or in $\mathcal{F}_i^{(j)}[k]$ (if $\tau_i^{(j)}[k]$ has been triggered), node i only adopts the information (based on (4.10)) of the neighbor u with the least delay. At this point, the delay of node i matches that of node u, and this fact is captured by the update rule (4.9). In case node i has no neighbor that has fresher information than itself w.r.t. sub-state j (where informativeness is quantified by $\tau_i^{(j)}[k]$), it increments its own freshness-index by 1 (as per (4.13)) to capture the effect of its own information getting older, and runs open-loop based on (4.12). Based on the above rules, at any given time-step k, $\tau_i^{(j)}[k]$ measures the age-of-information of $\hat{\mathbf{z}}_i^{(j)}[k]$, relative to the source node j. This fact is established later in Lemma 4.7.2. Finally, note that Algorithm 2 describes an approach for estimating $\mathbf{z}[k]$, and hence $\mathbf{x}[k]$, since $\mathbf{x}[k] = \mathbf{T}\mathbf{z}[k]$.

4.5 Performance Guarantees For Algorithm 2

4.5.1 Statement of the Results

In this section, we first state the theoretical guarantees afforded by Algorithm 2, and then discuss their implications; proofs of these statements are deferred to Section 4.7.1 and 4.7.2. The following is the main result of this chapter.

[7] Under Case 1 or Case 2 in Algo. 2, when a node $i \in \mathcal{V} \setminus \{j\}$ updates $\tau_i^{(j)}[k]$ via (4.9), and $\hat{\mathbf{z}}_i^{(j)}[k]$ via (4.10), we say that "i adopts the information of u at time k for sub-state j"; else, if it runs open-loop, we say it adopts its own information.

Theorem 4.5.1 *Given an LTI system (4.1), and a measurement model (4.2), suppose $(\mathbf{A}, \mathbf{C})$ is observable. Let the sequence of communication graphs $\{\mathcal{G}[k]\}_{k=0}^{\infty}$ satisfy conditions (C1)-(C3) in Section 4.2. Then, given any desired convergence rate $\rho \in (0,1)$, the observer gains $\mathbf{L}_1, \ldots, \mathbf{L}_N$ can be designed in a manner such that the estimation error of each node converges to zero exponentially fast at rate ρ, based on Algorithm 2.*

The next result states that under the conditions in Theorem 4.5.1, one can in fact achieve finite-time convergence.

Proposition 4.5.1 *(**Finite-Time Convergence**) Suppose the conditions stated in Theorem 4.5.1 hold. Then, the observer gains $\mathbf{L}_1, \ldots, \mathbf{L}_N$ can be designed in a manner such that the estimation error of each node converges to zero in finite-time.*

The next result follows directly from Prop. 4.5.1 and indicates that, when the sequence of communication graphs exhibits certain structure, one can derive a closed form expression for the maximum number of time-steps required for convergence.

Corollary 4.5.2 *Suppose the conditions stated in Theorem 4.5.1 hold. Additionally, suppose $f(t_q) \leq T, \forall t_q \in \mathbb{I}$, where $T \in \mathbb{N}_+$. Then, the observer gains $\mathbf{L}_1, \ldots, \mathbf{L}_N$ can be designed in a manner such that the estimation error of each node converges to zero in at most $n + 2N(N-1)T$ time-steps.*

Let us now discuss the implications of our results, and comment on certain aspects of our approach.

Remark 4.5.3 *The fact that a network of partially informed nodes can track the state of a dynamical system with arbitrarily large eigenvalues, over inter-communication intervals that can potentially grow unbounded, is non-obvious a priori. Our results in Theorem 4.5.1 and Proposition 4.5.1 indicate that not only can this be done exponentially fast at any desired rate, it can also be done in finite-time. In contrast, the result closest to ours [17] assumes strong-connectivity at each time-step - an assumption that is significantly stronger than what we make.*

Remark 4.5.4 *Presently, our approach requires a centralized design phase where the agents implement the multi-sensor decomposition in Section 4.2, and design their observer gains to achieve the desired convergence rate as outlined in the proof of Theorem 4.5.1. This is of course a limitation, but one that is common to all existing approaches [1, 17, 26, 30–32, 58, 102, 113–115] that we are aware of, i.e., each such approach involves a centralized design phase. Our approach also requires the nodes to know an upper-bound on the parameter δ in Eq.* (4.3) *while designing their observer gains. If, however, we constrain the inter-communication intervals to grow sub-linearly at worst, i.e, if $\delta = 0$, then such gains can be designed with no knowledge on the nature of the graph-sequences. Thus, there exists a trade-off between the generality of the graph sequences that can be tolerated, and the information required to do so.*

Remark 4.5.5 *When strong-connectivity is preserved over uniformly bounded intervals, i.e., when $f(t_q) \leq T, \forall t_q \in \mathbb{I}$, for some $T \in \mathbb{N}_+$, then our approach leads to bounded estimation errors under bounded disturbances. However, as we will see in Section 4.5.2, this may no longer be the case if the inter-communication intervals grow unbounded, no matter how slowly.*

4.5.2 Implications of growing inter-communication intervals under bounded disturbances

While Theorem 4.5.1 shows that estimation is possible even under growing inter-communication intervals, the goal of this section is to demonstrate via a simple example that this may no longer be true in the presence of disturbances. To this end, consider a scalar, unstable LTI system $x[k+1] = ax[k] + d$, where $a > 1$, and $d > 0$ is a disturbance input to the system. The network comprises of just 2 nodes: node 1 with measurement model $y_1[k] = c_1 x[k], c_1 \neq 0$, and node 2 with no measurements. Now consider an increasing sequence of time-steps $\mathbb{I} = \{t_0, t_1, \ldots\}$ with $t_0 = 0$, and let $f(t_q) = t_{q+1} - t_q, \forall t_q \in \mathbb{I}$ be a non-decreasing function of its argument, as in

Section 4.2. Suppose the communication pattern comprises of an edge from node 1 to node 2 precisely at the time-steps given by $\mathbb{I}$. Node 1 maintains a standard Luenberger observer given by $\hat{x}_1[k+1] = a\hat{x}_1[k] + l_1(y_1[k] - c_1\hat{x}_1[k])$, where l_1 is the observer gain. Node 2 applies Algorithm 2, which, in this case, translates to node 2 adopting the estimate of node 1 at each time-step t_q, and running open-loop at all other time-steps. Accordingly, we have $\hat{x}_2[t_{q+1}] = a^{f(t_q)}\hat{x}_1[t_q], \forall t_q \in \mathbb{I}$. With $e_i[k] = x[k] - \hat{x}_i[k], i \in \{1,2\}$, one can then easily verify:

$$\begin{aligned} e_1[k+1] &= \gamma e_1[k] + d, \forall k \in \mathbb{N}, \\ e_2[t_{q+1}] &= a^{f(t_q)} e_1[t_q] + d\frac{(a^{f(t_q)} - 1)}{(a-1)}, \forall t_q \in \mathbb{I}, \end{aligned} \tag{4.14}$$

where $\gamma = (a - l_1 c_1)$. Now consider a scenario where the inter-communication intervals grow unbounded, i.e., $f(t_q) \to \infty$ as $t_q \to \infty$. Since $a > 1$ and $d > 0$, it is clear from (4.14) that the error subsequence $e_2[t_q], t_q \in \mathbb{I}$ will grow unbounded even if node 1 chooses l_1 such that $\gamma = 0$. For the specific example under consideration, although the above arguments were constructed w.r.t. our algorithm, it seems unlikely that the final conclusion would change if one were to resort to some other approach.[8] The discussions in Section 4.5 can be thus summarized as follows.

- For a noiseless, disturbance free LTI system of the form (4.1), one can achieve exponential convergence at any desired rate, and even finite-time convergence based on Algorithm 2, under remarkably mild assumptions: joint observability, and joint strong-connectivity over intervals that can potentially grow unbounded.

- For an unstable system, any non-zero persistent disturbance, however small, can lead to unbounded estimation errors when the inter-communication intervals grow unbounded, no matter how slowly. Note however from Eq. (4.14) that our approach leads to bounded estimation errors under bounded disturbances if the sequence $\{f(t_q)\}_{t_q \in \mathbb{I}}$ is uniformly bounded above (see Remark 4.5.5).

[8] Note that we are only considering single-time-scale algorithms where nodes are not allowed to exchange their measurements. Also, we assume here that the nodes have no knowledge about the nature of the disturbance d, thereby precluding the use of any disturbance-rejection technique.

In light of the above points, the reasons for stating our results in full generality, i.e., for unbounded communication intervals, are as follows. First, we do so for theoretical interest, since we believe our work is the first to establish that the distributed state estimation problem can be solved with growing inter-communication intervals. Second, we essentially get this result for "free", i.e., accounting for such general scenarios incurs no additional steps in terms of the design of Algorithm 2. Finally, we emphasize that, while no existing approach can even handle the case where strong-connectivity is preserved over uniformly bounded time intervals (i.e., $\exists T \in \mathbb{N}_+$ such that $f(t_q) \leq T, \forall t_q \in \mathbb{I}$), the analysis for this scenario is simply a special case of that in Section 4.7.1.

4.6 Chapter Summary

We proposed a novel approach to the design of distributed observers for LTI systems over time-varying networks. We proved that our algorithm guarantees exponential convergence at any desired rate (including finite-time convergence) under graph-theoretic conditions that are far milder than those existing. In particular, we showed that these results hold even when the inter-communication intervals grow unbounded over time. To achieve our results, we made a connection to the concept of Age-of-Information (AoI) in the networking literature, and showed how it could be used to reject stale estimates of the state, thereby leading to strong theoretical guarantees for our problem.

4.7 Omitted Proofs

4.7.1 Proof of Theorem 4.5.1

The goal of this section is to prove Theorem 4.5.1. Before delving into the technical details, we first provide an informal discussion of the main ideas underlying the proof of Theorem 4.5.1. To this end, let us fix a sub-state $j \in \{1, \ldots, N\}$. The starting point of our analysis is Lemma 4.7.2 which establishes that for any non-source node $i \in \mathcal{V} \setminus \{j\}$, its error in estimation of sub-state j at time-step k can be expressed as a delayed version of the corresponding error of the source node j, where the delay is precisely the freshness-index $\tau_i^{(j)}[k]$. Given this result, we focus on bounding the delay $\tau_i^{(j)}[k]$ by exploiting the graph connectivity condition (C3). This is achieved in Lemma 4.7.3 where we first establish that $\tau_i^{(j)}[k]$ gets triggered after a finite period of time, and then show that it can be bounded above by the function $\tilde{g}(k) = 2(N-1)g(k)$, where $g(k)$ is as defined in Section 4.2. At this point, we appeal to condition (C2) (which caps the rate of growth of $\tilde{g}(k)$) in designing the observer gain $\mathbf{L}_j$ at node j. Specifically, in the proof of Theorem 4.5.1, we carefully design $\mathbf{L}_j$ such that despite a potentially growing delay, every non-source node $i \in \mathcal{V} \setminus \{j\}$ inherits the same exponential convergence to the true dynamics $\mathbf{z}^{(j)}[k]$ as that achieved by the corresponding source node j. With these ideas in place, we first prove a simple result that will be helpful later on; it states that a non-source node for a certain sub-state will always adopt the information of the corresponding source node, whenever it is in a position to do so.

Lemma 4.7.1 *Consider any sub-state j, and suppose that at some time-step k, we have $j \in \mathcal{N}_i[k]$, for some $i \in \mathcal{V} \setminus \{j\}$. Then, based on Algorithm 2, we have:*

(i) If $\tau_i^{(j)}[k] = \omega$, then $j = \operatorname{argmin}_{l \in \mathcal{M}_i^{(j)}[k]} \tau_l^{(j)}[k]$.

(ii) If $\tau_i^{(j)}[k] \neq \omega$, then $j = \operatorname{argmin}_{l \in \mathcal{F}_i^{(j)}[k]} \tau_l^{(j)}[k]$.

Proof The result follows from two simple observations that are direct consequences of the rules of Algorithm 2: (i) $\tau_j^{(j)}[k] = 0, \forall k \in \mathbb{N}$, and (ii) for any $i \in \mathcal{V} \setminus \{j\}$, $\tau_i^{(j)}[k] \geq 1$ whenever $\tau_i^{(j)}[k] \neq \omega$. In other words, the source node for a given sub-state has the lowest freshness-index for that sub-state at all time-steps. ■

Lemma 4.7.2 *Suppose all nodes employ Algorithm 2. Consider any sub-state j, and suppose that at some time-step k, we have $\tau_i^{(j)}[k] = m$, where $i \in \mathcal{V} \setminus \{j\}$, and $m \in \mathbb{N}_+$.*

Then, there exist nodes $v(\tau) \in \mathcal{V} \setminus \{j\}, \tau \in \{k-m+1, \ldots, k\}$, such that the following is true:

$$\hat{\mathbf{z}}_i^{(j)}[k] = \mathbf{A}_{jj}^m \hat{\mathbf{z}}_j^{(j)}[k-m] + \sum_{q=1}^{(j-1)} \sum_{\tau=(k-m)}^{(k-1)} \mathbf{A}_{jj}^{(k-\tau-1)} \mathbf{A}_{jq} \hat{\mathbf{z}}_{v(\tau+1)}^{(q)}[\tau]. \tag{4.15}$$

Proof Consider any sub-state j, and suppose that at some time-step k, we have $\tau_i^{(j)}[k] = m$, where $i \in \mathcal{V} \setminus \{j\}$, and $m \in \mathbb{N}_+$. Given this scenario, we claim that there exist nodes $v(\tau) \in \mathcal{V} \setminus \{j\}, \tau \in \{k-m+1, \ldots, k\}$ such that $v(\tau)$ adopts the information of $v(\tau-1)$ at time $\tau - 1$ for sub-state j, $\forall \tau \in \{k-m+1, \ldots, k\}$, with $v(k-m) = j$ and $v(k) = i$. As we shall see, establishing this claim readily establishes (4.15); thus, we first focus on proving the former via induction on m. For the base case of induction, suppose $\tau_i^{(j)}[k] = 1$ for some $i \in \mathcal{V} \setminus \{j\}$ at some time-step k. Based on Algorithm 2 and Lemma 4.7.1, note that this is possible if and only if $j \in \mathcal{N}_i[k-1]$. In particular, $v(k) = i$ would then adopt the information of $v(k-1) = j$ at time $k-1$ for sub-state j. This establishes the claim when $m = 1$. Now fix an integer $r \geq 2$, and suppose the claim is true for all $m \in \{1, \ldots, r-1\}$. Suppose $\tau_i^{(j)}[k] = r$ for some $i \in \mathcal{V} \setminus \{j\}$ at some time-step k. From Algorithm 2, observe that this is true if and only if i adopts the information of some node $l \in \mathcal{N}_i[k-1] \cup \{i\}$ at time $k-1$ for sub-state j, such that $\tau_l^{(j)}[k-1] = r-1$. Since $r - 1 \geq 1$, it must be that $l \in \mathcal{V} \setminus \{j\}$; the induction hypothesis thus applies to node l. Using this fact, and setting $v(k-1) = l$ completes our inductive proof of the claim. Finally, observe that for any $\tau \in \{k-m+1, \ldots, k\}$, whenever $v(\tau)$ adopts the information of $v(\tau-1)$ at $\tau - 1$, the following identity holds based on (4.10) and (4.12):

$$\hat{\mathbf{z}}_{v(\tau)}^{(j)}[\tau] = \mathbf{A}_{jj} \hat{\mathbf{z}}_{v(\tau-1)}^{(j)}[\tau-1] + \sum_{q=1}^{(j-1)} \mathbf{A}_{jq} \hat{\mathbf{z}}_{v(\tau)}^{(q)}[\tau-1]. \tag{4.16}$$

Using the above identity repeatedly for all $\tau \in \{k-m+1, \ldots, k\}$ with $v(k-m) = j$ and $v(k) = i$, immediately leads to (4.15). This completes the proof. ■

Lemma 4.7.3 *Suppose the sequence $\{\mathcal{G}[k]\}_{k=0}^{\infty}$ satisfies condition (C3) in Section 4.2. Then, for each sub-state j, Algorithm 2 guarantees the following.*

$$\tau_i^{(j)}[k] \neq \omega, \forall k \geq \sum_{q=0}^{N-2} f(t_q), \forall i \in \mathcal{V}, \text{ and} \tag{4.17}$$

$$\tau_i^{(j)}[t_{p(N-1)}] \leq \sum_{q=(p-1)(N-1)}^{p(N-1)-1} f(t_q), \forall p \in \mathbb{N}_+, \forall i \in \mathcal{V}. \tag{4.18}$$

Proof Fix a sub-state j, and notice that both (4.17) and (4.18) hold for the corresponding source node j, since $\tau_j^{(j)}[k] = 0, \forall k \in \mathbb{N}$. To establish these claims for the remaining nodes, we begin by making the following simple observation that follows directly from (4.9) and (4.13), and applies to every node $i \in \mathcal{V} \setminus \{j\}$:

$$\tau_i^{(j)}[k+1] \leq \tau_i^{(j)}[k] + 1, \text{ whenever } \tau_i^{(j)}[k] \neq \omega. \tag{4.19}$$

Our immediate goal is to establish (4.18) when $p = 1$ and, in the process, establish (4.17). Let $\mathcal{C}_0^{(j)} = \{j\}$, and define:

$$\mathcal{C}_1^{(j)} \triangleq \{i \in \mathcal{V} \setminus \mathcal{C}_0^{(j)} : \{\bigcup_{\tau=t_0}^{t_1-1} \mathcal{N}_i[\tau]\} \cap \mathcal{C}_0^{(j)} \neq \emptyset\}. \tag{4.20}$$

In words, $\mathcal{C}_1^{(j)}$ represents the set of non-source nodes (for sub-state j) that have a direct edge from node j at least once over the interval $[t_0, t_1)$. Based on condition (C3), $\mathcal{C}_1^{(j)}$ is non-empty (barring the trivial case when $\mathcal{V} = \{j\}$). For each $i \in \mathcal{C}_1^{(j)}$, it must be that $j \in \mathcal{M}_i^{(j)}[\bar{k}]$ for some $\bar{k} \in [t_0, t_1)$. Thus, based on (4.9) and (4.13), it must be that $\tau_i^{(j)}[k] \neq \omega, \forall k \geq t_1 = f(t_0), \forall i \in \mathcal{C}_1^{(j)}$. In particular, we note based on (4.19) that $\tau_i^{(j)}[t_1] \leq t_1$, and hence $\tau_i^{(j)}[t_{N-1}] \leq t_{N-1} = \sum_{q=0}^{N-2} f(t_q), \forall i \in \mathcal{C}_1^{(j)}$. We can keep repeating the above argument by recursively defining the sets $\mathcal{C}_r^{(j)}, 1 \leq r \leq (N-1)$, as follows:

$$\mathcal{C}_r^{(j)} \triangleq \{i \in \mathcal{V} \setminus \bigcup_{q=0}^{(r-1)} \mathcal{C}_q^{(j)} : \{\bigcup_{\tau=t_{r-1}}^{t_r-1} \mathcal{N}_i[\tau]\} \cap \{\bigcup_{q=0}^{(r-1)} \mathcal{C}_q^{(j)}\} \neq \emptyset\}. \tag{4.21}$$

We proceed via induction on r. Suppose the following is true for all $r \in \{1, \ldots, m-1\}$, where $m \in \{2, \ldots, N-1\}$: $\tau^{(j)}_i[t_r] \neq \omega$ and $\tau^{(j)}_i[t_r] \leq t_r, \forall i \in \bigcup_{q=0}^{r} \mathcal{C}^{(j)}_q$. Now suppose $r = m$. If $\mathcal{V} \setminus \bigcup_{q=0}^{(m-1)} \mathcal{C}^{(j)}_q$ is empty, then we are done establishing (4.17), and (4.18) for the case when $p = 1$. Else, based on condition (C3), it must be that $\mathcal{C}^{(j)}_m$ is non-empty. Consider a node $i \in \mathcal{C}^{(j)}_m$. Based on the way $\mathcal{C}^{(j)}_m$ is defined, note that at some time-step $\bar{k} \in [t_{m-1}, t_m)$, node i has some neighbor v (say) from the set $\bigcup_{q=0}^{(m-1)} \mathcal{C}^{(j)}_q$. Based on the induction hypothesis and (4.19), it must be that $\tau^{(j)}_v[\bar{k}] \neq \omega$ and $\tau^{(j)}_v[\bar{k}] \leq \bar{k}$. At this point, if $\tau^{(j)}_i[\bar{k}] = \omega$, then since $v \in \mathcal{M}^{(j)}_i[\bar{k}]$, node i would update $\tau^{(j)}_i[\bar{k}]$ based on (4.9). Else, if $\tau^{(j)}_i[\bar{k}] \neq \omega$, there are two possibilities: (i) $v \in \mathcal{F}^{(j)}_i[\bar{k}]$, implying $\mathcal{F}^{(j)}_i[\bar{k}] \neq \emptyset$; or (ii) $v \notin \mathcal{F}^{(j)}_i[\bar{k}]$, implying $\tau^{(j)}_i[\bar{k}] \leq \tau^{(j)}_v[\bar{k}] \leq \bar{k}$. The above discussion, coupled with the freshness-index update rules for Case 2 of the algorithm (line 5 of Algo. 2), and (4.19), imply $\tau^{(j)}_i[t_m] \neq \omega$ and $\tau^{(j)}_i[t_m] \leq t_m$. This completes the induction step. Appealing to (4.19) once again, and noting that $\bigcup_{q=0}^{N-1} \mathcal{C}^{(j)}_q = \mathcal{V}$ and $t_{N-1} = \sum_{q=0}^{N-2} f(t_q)$, establishes (4.17), and (4.18) when $p = 1$.

In order to establish (4.18) for any $p \in \mathbb{N}_+$, one can follow a similar line of argument as above to analyze the evolution of the freshness indices over the interval $[t_{(p-1)(N-1)}, t_{p(N-1)}]$. In particular, for any $p > 1$, we can set $\mathcal{D}^{(j)}_0 = \{j\}$, and define the sets $\mathcal{D}^{(j)}_r, 1 \leq r \leq (N-1)$ recursively as follows:

$$\mathcal{D}^{(j)}_r \triangleq \{i \in \mathcal{V} \setminus \bigcup_{q=0}^{(r-1)} \mathcal{D}^{(j)}_q : \{ \bigcup_{\tau=t_{h(p,N,r)-1}}^{t_{h(p,N,r)}-1} \mathcal{N}_i[\tau]\} \cap \{ \bigcup_{q=0}^{(r-1)} \mathcal{D}^{(j)}_q\} \neq \emptyset\}, \tag{4.22}$$

where $h(p, N, r) = (p-1)(N-1) + r$. One can then establish that $\tau^{(j)}_i[t_{h(p,N,r)}] \leq \sum_{q=(p-1)(N-1)}^{h(p,N,r)-1} f(t_q), \forall i \in \mathcal{D}^{(j)}_r$, $\forall r \in \{1, \ldots, N-1\}$, via induction. ■

We are now in position to prove Theorem 4.5.1.

Proof (Theorem 4.5.1) The proof is divided into two parts. In the first part, we describe a procedure for designing the observer gains $\{\mathbf{L}_i\}_{i=1}^{N}$. In the second part, we establish that our choice of observer gains indeed leads to the desired exponential convergence rate ρ.

Design of the observer gains: We begin by noting that for each sub-state j, one can always find scalars $\beta_j, \gamma_j \geq 1$, such that $\left\|(\mathbf{A}_{jj})^k\right\| \leq \beta_j \gamma_j^k, \forall k \in \mathbb{N}$ [105].[9] Define $\gamma \triangleq \max\limits_{1\leq j\leq N} \gamma_j$. Next, fix a $\bar{\delta} \in (\delta, 1)$, where δ is as in (4.3). Given a desired rate of convergence $\rho \in (0,1)$, we now recursively define two sets of positive scalars, namely $\{\rho_j\}_{j=1}^N$ and $\{\lambda_j\}_{j=1}^N$, starting with $j = N$. With $\lambda_N = \rho$, let $\rho_j, j = N$, be chosen to satisfy:

$$\gamma^{\bar{\delta}} \rho_j^{1-\bar{\delta}} \leq \lambda_j. \tag{4.23}$$

Having picked $\rho_j \in (0,1)$ to meet the above condition, we set λ_{j-1} to be any number in $(0, \rho_j)$, pick ρ_{j-1} to satisfy (4.23), and then repeat this process till we reach $j = 1$. Observe that the sets $\{\rho_j\}_{j=1}^N$ and $\{\lambda_j\}_{j=1}^N$ as defined above always exist, and satisfy: $\rho_1 < \lambda_1 < \rho_2 < \lambda_2 < \cdots < \lambda_{N-1} < \rho_N < \lambda_N = \rho$. For each sub-state $j \in \{1, \ldots, N\}$, let the corresponding source node j design the observer gain $\mathbf{L}_j$ (featuring in equation (4.8)) in a manner such that the matrix $(\mathbf{A}_{jj} - \mathbf{L}_j\mathbf{C}_{jj})$ has distinct real eigenvalues with spectral radius equal to ρ_j. Such a choice of $\mathbf{L}_j$ exists as the pair $(\mathbf{A}_{jj}, \mathbf{C}_{jj})$ is observable by construction. This completes our design procedure.

Convergence analysis: We first note that there exists a set of positive scalars $\{\alpha_1, \ldots, \alpha_N\}$, such that [105]:

$$\left\|(\mathbf{A}_{jj} - \mathbf{L}_j\mathbf{C}_{jj})^k\right\| \leq \alpha_j \rho_j^k, \forall k \in \mathbb{N}. \tag{4.24}$$

For a particular sub-state j, let $\mathbf{e}_i^{(j)}[k] = \hat{\mathbf{z}}_i^{(j)}[k] - \mathbf{z}^{(j)}[k]$. Consider the first sub-state $j = 1$, and observe that based on (4.4), (4.5), and (4.8), the following is true: $\mathbf{e}_1^{(1)}[k+1] = (\mathbf{A}_{11} - \mathbf{L}_1\mathbf{C}_{11})\mathbf{e}_1^{(1)}[k]$. Thus, we obtain

$$\mathbf{e}_1^{(1)}[k] = (\mathbf{A}_{11} - \mathbf{L}_1\mathbf{C}_{11})^k \mathbf{e}_1^{(1)}[0]. \tag{4.25}$$

Based on (4.24) and (4.25), we then have:

$$\left\|\mathbf{e}_1^{(1)}[k]\right\| \leq c_1 \rho_1^k, \forall k \in \mathbb{N}, \tag{4.26}$$

where $c_1 \triangleq \alpha_1 \left\|\mathbf{e}_1^{(1)}[0]\right\|$. Given that node 1's error for sub-state 1 decays exponentially as per (4.26), we want to now relate the errors $\mathbf{e}_i^{(1)}[k], i \in \mathcal{V} \setminus \{1\}$ of the non-source

[9] We use $\|\mathbf{A}\|$ to refer to the induced 2-norm of a matrix $\mathbf{A}$.

nodes (for sub-state 1) to $\mathbf{e}_1^{(1)}[k]$. To this end, consider any $i \in \mathcal{V} \setminus \{1\}$, and note that for any $k \geq t_{N-1}$, Eq. (4.17) in Lemma 4.7.3 implies that $\tau_i^{(1)}[k] \neq \omega$, and hence $\tau_i^{(1)}[k] \in \mathbb{N}_+$. Invoking Lemma 4.7.2, and using the fact that $\mathbf{z}^{(1)}[k] = (\mathbf{A}_{11})^m \mathbf{z}^{(1)}[k-m], \forall m \in \mathbb{N}$, we then obtain the following $\forall i \in \mathcal{V} \setminus \{1\}$:

$$\mathbf{e}_i^{(1)}[k] = (\mathbf{A}_{11})^{\tau_i^{(1)}[k]} \mathbf{e}_1^{(1)}[k - \tau_i^{(1)}[k]], \forall k \geq t_{N-1}. \tag{4.27}$$

Our next goal is to bound the delay term $\tau_i^{(1)}[k]$ in the above relation. For this purpose, consider any time-step $k \geq t_{N-1}$, and let $p(k)$ be the largest integer such that $t_{p(k)(N-1)} \leq k$. Then, for any sub-state j, and any $i \in \mathcal{V} \setminus \{j\}$, we observe:

$$\begin{aligned} \tau_i^{(j)}[k] &\overset{(a)}{\leq} \tau_i^{(j)}[t_{p(k)(N-1)}] + (k - t_{p(k)(N-1)}) \\ &\overset{(b)}{\leq} \sum_{q=(p(k)-1)(N-1)}^{p(k)(N-1)-1} f(t_q) + (k - t_{p(k)(N-1)}) \\ &\overset{(c)}{\leq} 2(N-1) f(m(k)) \overset{(d)}{=} 2(N-1) g(k). \end{aligned} \tag{4.28}$$

In the above inequalities, (a) follows from (4.17) in Lemma 4.7.3 and (4.19); (b) follows from (4.18) in Lemma 4.7.3; and (c) follows from the monotonicity of $f(\cdot)$ in condition (C1), and by recalling that $m(k) \triangleq \max\{t_q \in \mathbb{I} : t_q \leq k\}$. Finally, (d) follows by recalling that $g(k) = f(m(k))$. Recalling that $\left\|(\mathbf{A}_{11})^k\right\| \leq \beta_1 \gamma_1^k$, using the bounds in (4.26) and (4.28), the fact that $\gamma_1 \geq 1$ and $\rho_1 < 1$, and the sub-multiplicative property of the 2-norm, we obtain the following by taking norms on both sides of (4.27):

$$\left\|\mathbf{e}_i^{(1)}[k]\right\| \leq \bar{c}_1 \left(\frac{\gamma_1}{\rho_1}\right)^{\tilde{g}(k)} \rho_1^k, \forall k \geq t_{N-1}, \forall i \in \mathcal{V} \setminus \{1\}, \tag{4.29}$$

where $\tilde{g}(k) = 2(N-1)g(k)$ and $\bar{c}_1 \triangleq c_1 \beta_1$. Based on condition (C3), and our choice of $\bar{\delta}$, observe that there exists $\bar{k}(\bar{\delta})$ such that $\tilde{g}(k) \leq \bar{\delta} k, \forall k \geq \bar{k}(\bar{\delta})$. With $k_1 \triangleq \max\{t_{N-1}, \bar{k}(\bar{\delta})\}$, we then obtain the following based on (4.23) and (4.29), for all $k \geq k_1$ and for all $i \in \mathcal{V} \setminus \{1\}$:

$$\left\|\mathbf{e}_i^{(1)}[k]\right\| \leq \bar{c}_1 \left(\gamma_1^{\bar{\delta}} \rho_1^{1-\bar{\delta}}\right)^k \leq \bar{c}_1 \left(\gamma^{\bar{\delta}} \rho_1^{1-\bar{\delta}}\right)^k \leq \bar{c}_1 \lambda_1^k. \tag{4.30}$$

Note that since $\bar{c}_1 \geq c_1$ and $\lambda_1 > \rho_1$, the above bound applies to node 1 as well (see equation (4.26)). We have thus established that exponential convergence at rate λ_1 for sub-state 1 holds for each node in the network.

Our aim is to now obtain a bound similar to that in (4.30) for each sub-state $j \in \{2, \ldots, N\}$. To this end, with $g_{jq} = \|(\mathbf{A}_{jq} - \mathbf{L}_j \mathbf{C}_{jq})\|$ and $h_{jq} = \|\mathbf{A}_{jq}\|$, let us define the following quantities recursively for $j \in \{2, \ldots, N\}$:

$$\begin{aligned} k_j &\triangleq \frac{k_{j-1}}{(1-\bar{\delta})}, \\ c_j &\triangleq \frac{\alpha_j}{\rho_j^{k_{j-1}}} \left(\left\| \mathbf{e}_j^{(j)}[k_{j-1}] \right\| + \sum_{q=1}^{(j-1)} \frac{g_{jq}\bar{c}_q}{(\rho_j - \lambda_q)} \lambda_q^{k_{j-1}} \right), \\ \bar{c}_j &\triangleq \beta_j \left(c_j + \sum_{q=1}^{(j-1)} \frac{h_{jq}\bar{c}_q}{(\gamma_j - \lambda_q)} \right), \end{aligned} \tag{4.31}$$

where $k_1 \triangleq \max\{t_{N-1}, \bar{k}(\bar{\delta})\}$, $c_1 \triangleq \alpha_1 \left\| \mathbf{e}_1^{(1)}[0] \right\|$, and $\bar{c}_1 = c_1 \beta_1$. Based on the above definitions, we claim that for each sub-state $j \in \{1, \ldots, N\}$, the following is true:

$$\left\| \mathbf{e}_i^{(j)}[k] \right\| \leq \bar{c}_j \lambda_j^k, \forall k \geq k_j, \forall i \in \mathcal{V}. \tag{4.32}$$

We will prove the above claim via induction on the sub-state number j. We have already established (4.32) for the base case when $j = 1$. For $j \geq 2$, our strategy will be to first analyze the evolution of $\mathbf{e}_j^{(j)}[k]$ at the source node j. From (4.4) and (4.5), we note that the dynamics of the j-th sub-state are coupled with those of the first $j-1$ sub-states. Thus, $\mathbf{e}_j^{(j)}[k]$ will exhibit exponential decay only when the errors for the first $j-1$ sub-states have already started decaying exponentially, with k_{j-1} (as defined in (4.31)) representing the instant when exponential decay for the $(j-1)$-th sub-state kicks in. Let us now prove that as soon as this happens, the following holds:

$$\left\| \mathbf{e}_j^{(j)}[k] \right\| \leq c_j \rho_j^k, \forall k \geq k_{j-1}. \tag{4.33}$$

To do so, suppose (4.32) holds for all $j \in \{1, \ldots, l-1\}$, where $l \in \{2, \ldots, N\}$. Now let $j = l$ and observe that equations (4.4) and (4.5) yield:

$$\begin{aligned}\mathbf{z}^{(l)}[k+1] &= \mathbf{A}_{ll}\mathbf{z}^{(l)}[k] + \sum_{q=1}^{(l-1)} \mathbf{A}_{lq}\mathbf{z}^{(q)}[k] \\ &= (\mathbf{A}_{ll} - \mathbf{L}_l\mathbf{C}_{ll})\mathbf{z}^{(l)}[k] + \sum_{q=1}^{(l-1)} (\mathbf{A}_{lq} - \mathbf{L}_l\mathbf{C}_{lq})\mathbf{z}^{(q)}[k] + \mathbf{L}_l\mathbf{y}_l[k].\end{aligned} \tag{4.34}$$

Based on the above equation and (4.8), we obtain:

$$\mathbf{e}_l^{(l)}[k+1] = (\mathbf{A}_{ll} - \mathbf{L}_l\mathbf{C}_{ll})\mathbf{e}_l^{(l)}[k] + \sum_{q=1}^{(l-1)} (\mathbf{A}_{lq} - \mathbf{L}_l\mathbf{C}_{lq})\mathbf{e}_l^{(q)}[k].$$

Rolling out the above equation starting from k_{l-1} yields:

$$\mathbf{e}_l^{(l)}[k] = {F_{ll}}^{(k-k_{l-1})}\mathbf{e}_l^{(l)}[k_{l-1}] + \sum_{q=1}^{(l-1)} \sum_{\tau=k_{l-1}}^{(k-1)} {F_{ll}}^{(k-\tau-1)} G_{lq}\mathbf{e}_l^{(q)}[\tau], \tag{4.35}$$

where $F_{ll} = (\mathbf{A}_{ll} - \mathbf{L}_l\mathbf{C}_{ll})$, and $G_{lq} = (\mathbf{A}_{lq} - \mathbf{L}_l\mathbf{C}_{lq})$. Taking norms on both sides of the above equation, using the triangle inequality, and the sub-multiplicative property of the two-norm, we obtain:

$$\begin{aligned}\left\|\mathbf{e}_l^{(l)}[k]\right\| &\overset{(a)}{\leq} \alpha_l\rho_l^k \left(\frac{\left\|\mathbf{e}_l^{(l)}[k_{l-1}]\right\|}{\rho_l^{k_{l-1}}} + \frac{1}{\rho_l}\sum_{q=1}^{(l-1)} g_{lq} \sum_{\tau=k_{l-1}}^{(k-1)} \rho_l^{-\tau} \left\|\mathbf{e}_l^{(q)}[\tau]\right\| \right) \\ &\overset{(b)}{\leq} \alpha_l\rho_l^k \left(\frac{\left\|\mathbf{e}_l^{(l)}[k_{l-1}]\right\|}{\rho_l^{k_{l-1}}} + \frac{1}{\rho_l}\sum_{q=1}^{(l-1)} g_{lq}\bar{c}_q \sum_{\tau=k_{l-1}}^{\infty} \left(\frac{\lambda_q}{\rho_l}\right)^{\tau} \right) \\ &\overset{(c)}{\leq} c_l\rho_l^k, \forall k \geq k_{l-1}.\end{aligned} \tag{4.36}$$

In the above inequalities, (a) follows from (4.24) and by recalling that $g_{lq} = \|G_{lq}\|$; (b) follows by first applying the induction hypothesis noting that $q \leq (l-1)$ and $\tau \geq k_{l-1}$, and then changing the upper limit of the inner summation (over time); (c) follows by simplifying the preceding inequality using the fact that $\lambda_q < \rho_l, \forall q \in \{1, \ldots, l-1\}$, and using the definition of c_l in (4.31). We have thus obtained a bound on the estimation error of sub-state l at node l. To obtain a similar bound for each $i \in \mathcal{V}\setminus\{l\}$, note that equation (4.34) can be rolled out over time to yield the following for each $m \in \mathbb{N}$:

$$\mathbf{z}^{(l)}[k] = \mathbf{A}_{ll}^{m}\mathbf{z}^{(l)}[k-m] + \sum_{q=1}^{(l-1)} \sum_{\tau=(k-m)}^{(k-1)} \mathbf{A}_{ll}^{(k-\tau-1)}\mathbf{A}_{lq}\mathbf{z}^{(q)}[\tau].$$

Leveraging Lemma 4.7.2, we can then obtain the following error dynamics for a node $i \in \mathcal{V} \setminus \{l\}, \forall k \geq t_{N-1}$.

$$\begin{aligned}\mathbf{e}_i^{(l)}[k] &= (\mathbf{A}_{ll})^{\tau_i^{(l)}[k]}\mathbf{e}_l^{(l)}[k-\tau_i^{(l)}[k]] \\ &+ \sum_{q=1}^{(l-1)} \sum_{\tau=(k-\tau_i^{(l)}[k])}^{(k-1)} \mathbf{A}_{ll}^{(k-\tau-1)}\mathbf{A}_{lq}\mathbf{e}_{v(\tau+1)}^{(q)}[\tau].\end{aligned} \tag{4.37}$$

Based on the above equation, we note that since $\mathbf{A}_{ll}$ can contain unstable eigenvalues, and since $\tau_i^{(l)}[k]$ may grow over time (owing to potentially growing inter-communication intervals), we need the decay in $\mathbf{e}_l^{(l)}[k-\tau_i^{(l)}[k]]$ to dominate the growth due to $(\mathbf{A}_{ll})^{\tau_i^{(l)}[k]}$ in order for $\mathbf{e}_i^{(l)}[k]$ to eventually remain bounded. To show that this is indeed the case, we begin by noting the following inequalities that hold when $k \geq k_l$:

$$\frac{k_{l-1}}{k} \overset{(a)}{\leq} 1 - \bar{\delta} \overset{(b)}{\leq} 1 - \frac{\tilde{g}(k)}{k} \overset{(c)}{\leq} 1 - \frac{\tau_i^{(l)}[k]}{k}, \tag{4.38}$$

where $\tilde{g}(k) = 2(N-1)g(k)$. In the above inequalities, (a) follows directly from (4.31); (b) follows by noting that $k \geq k_l \implies k \geq \bar{k}(\bar{\delta})$; and (c) follows from (4.28) and by noting that $k \geq k_l \implies k \geq t_{N-1}$. We conclude that if $k \geq k_l$, then $k - \tau_i^{(l)}[k] \geq k_{l-1}$. Thus, when $k \geq k_l$, at any time-step $\tau \geq k - \tau_i^{(l)}[k]$, the errors of the first $l-1$ sub-states would exhibit exponential decay based on the induction hypothesis. With this in mind, we fix $i \in \mathcal{V} \setminus \{l\}$, $k \geq k_l$, and bound $\mathbf{e}_i^{(l)}[k]$ by taking norms on both sides of (4.37), as follows:

$$\begin{aligned}
\left\|\mathbf{e}_i^{(l)}[k]\right\| &\overset{(a)}{\leq} \beta_l \left(c_l \left(\frac{\gamma_l}{\rho_l}\right)^{\tilde{g}(k)} \rho_l^k + \gamma_l^{(k-1)} \sum_{q=1}^{(l-1)} h_{lq}\bar{c}_q \sum_{\tau=(k-\tau_i^{(l)}[k])}^{(k-1)} \left(\frac{\lambda_q}{\gamma_l}\right)^{\tau} \right) \\
&\overset{(b)}{\leq} \beta_l \left(c_l \left(\frac{\gamma_l}{\rho_l}\right)^{\tilde{g}(k)} \rho_l^k + \gamma_l^{(k-1)} \sum_{q=1}^{(l-1)} h_{lq}\bar{c}_q \sum_{\tau=(k-\tilde{g}(k))}^{\infty} \left(\frac{\lambda_q}{\gamma_l}\right)^{\tau} \right) \\
&\overset{(c)}{=} \beta_l \left(c_l \left(\frac{\gamma_l}{\rho_l}\right)^{\tilde{g}(k)} \rho_l^k + \sum_{q=1}^{(l-1)} \frac{h_{lq}\bar{c}_q}{(\gamma_l - \lambda_q)} \left(\frac{\gamma_l}{\lambda_q}\right)^{\tilde{g}(k)} \lambda_q^k \right) \\
&\overset{(d)}{\leq} \bar{c}_l \left(\gamma^{\bar{\delta}} \rho_l^{1-\bar{\delta}}\right)^k \overset{(e)}{\leq} \bar{c}_l \lambda_l^k .
\end{aligned} \tag{4.39}$$

In the above steps, (a) follows by first recalling that $\left\|(\mathbf{A}_{ll})^k\right\| \leq \beta_l \gamma_l^k, \forall k \in \mathbb{N}$, $h_{lq} = \|\mathbf{A}_{lq}\|$, $\tilde{g}(k) = 2(N-1)g(k)$, and then using the induction hypothesis, equations (4.28), (4.32), and (4.36), and the facts that $\rho_l < 1, \gamma_l \geq 1$; (b) follows by suitably changing the upper and lower limits of the inner summation (over time), a change that is warranted since each summand is non-negative; (c) follows by simplifying the preceding inequality; (d) follows by noting that $\lambda_q < \rho_l, \forall q \in \{1, \ldots, l-1\}$, using the definition of $\bar{c}_l$ in (4.31), and the fact that $\tilde{g}(k) \leq \bar{\delta}k, \forall k \geq k_l$; and finally (e) follows from (4.23). This completes the induction step. Let $\mathbf{e}_i[k] = \hat{\mathbf{z}}_i[k] - \mathbf{z}[k]$. Recalling that $\lambda_j \leq \rho, \forall j \in \{1, \ldots, N\}$, we obtain as desired:

$$\|\mathbf{e}_i[k]\| = \sqrt{\sum_{j=1}^{N} \left\|\mathbf{e}_i^{(j)}[k]\right\|^2} \leq \left(\sqrt{\sum_{j=1}^{N} \bar{c}_j^2}\right) \rho^k, \forall k \geq k_N, \forall i \in \mathcal{V}.$$

■

4.7.2 Proof of Proposition 4.5.1

Proof (Proposition 4.5.1) For each sub-state $j \in \{1, \ldots, N\}$, let the corresponding source node j design the observer gain $\mathbf{L}_j$ (featuring in equation (4.8)) in a manner such that the matrix $(\mathbf{A}_{jj} - \mathbf{L}_j\mathbf{C}_{jj})$ has all its eigenvalues at 0. Such a choice of $\mathbf{L}_j$

exists based on the fact that the pair $(\mathbf{A}_{jj}, \mathbf{C}_{jj})$ is observable by construction. Let $n_j = dim(\mathbf{A}_{jj})$. Given the above choice of observer gains, we will prove the result by providing an upper bound on the number of time-steps it takes the error of each node to converge to $\mathbf{0}$. To this end, we first define a sequence $\{\bar{\tau}_j\}_{j=1}^{N}$ of time-steps as follows:

$$\bar{\tau}_1 = \inf\{\tau \in \mathbb{N}_+, \tau \geq t_{N-1} : k - \tilde{g}(k) \geq n_1, \forall k \geq \tau\}, \tag{4.40}$$

where $\tilde{g}(k) = 2(N-1)g(k)$, and

$$\bar{\tau}_j = \inf\{\tau \in \mathbb{N}_+ : k - \tilde{g}(k) \geq \bar{\tau}_{j-1} + n_j, \forall k \geq \tau\}. \tag{4.41}$$

Based on condition (C3), namely $\limsup_{k\to\infty} g(k)/k = \delta < 1$, observe that $\bar{\tau}_j$ as defined above is finite $\forall j \in \{1, \ldots, N\}$. Next, note that by construction, $(\mathbf{A}_{jj} - \mathbf{L}_j \mathbf{C}_{jj})$ is a nilpotent matrix of index at most n_j. Thus, it is easy to see that $\mathbf{e}_1^{(1)}[k] = \mathbf{0}, \forall k \geq n_1$, based on (4.25). Recall from (4.28) that for each sub-state j, $\tau_i^{(j)}[k] \leq \tilde{g}(k), \forall k \geq t_{N-1}, \forall i \in \mathcal{V}$. From the definition of $\bar{\tau}_1$ in (4.40), and equation (4.27), we immediately obtain that $\mathbf{e}_i^{(1)}[k] = \mathbf{0}, \forall k \geq \bar{\tau}_1, \forall i \in \mathcal{V}$. One can easily generalize this argument to the remaining sub-states by using an inductive reasoning akin to that in the proof of Theorem 4.5.1. In particular, for any sub-state $j \in \{2, \ldots, N\}$, one can roll out the error dynamics for node j as in (4.35), starting from time-step $\bar{\tau}_{j-1}$. By this time, the induction hypothesis would imply that the estimation errors of all nodes on all sub-states $q \in \{1, \ldots, j-1\}$ have converged to zero. The nilpotentcy of $(\mathbf{A}_{jj} - \mathbf{L}_j \mathbf{C}_{jj})$ would then imply that $\mathbf{e}_j^{(j)}[k] = \mathbf{0}, \forall k \geq \bar{\tau}_{j-1} + n_j$. From the definition of $\bar{\tau}_j$ in (4.41), and (4.28), we note that $k \geq \bar{\tau}_j \implies k - \tau_i^{(j)}[k] \geq \bar{\tau}_{j-1} + n_j, \forall i \in \mathcal{V}$. Referring to (4.37), we conclude that $\mathbf{e}_i^{(j)}[k] = \mathbf{0}, \forall k \geq \bar{\tau}_j, \forall i \in \mathcal{V}$. Based on the above reasoning, the overall error for each node converges to $\mathbf{0}$ in at most $\bar{\tau}_N$ time-steps. ■

Part II

Distributed Hypothesis Testing and Non-Bayesian Learning over Networks

5. A NEW APPROACH TO DISTRIBUTED HYPOTHESIS TESTING AND NON-BAYESIAN LEARNING: IMPROVED LEARNING RATE AND BYZANTINE-RESILIENCE

In this chapter, we study a setting where a group of agents, each receiving partially informative private signals, seek to collaboratively learn the true underlying state of the world (from a finite set of hypotheses) that generates their joint observation profiles. To solve this problem, we propose a distributed learning rule that differs fundamentally from existing approaches, in that it does not employ any form of "belief-averaging". Instead, agents update their beliefs based on a min-rule. Under standard assumptions on the observation model and the network structure, we establish that each agent learns the truth asymptotically almost surely. As our main contribution, we prove that with probability 1, each false hypothesis is ruled out by every agent exponentially fast, at a network-independent rate that is strictly larger than existing rates. We then develop a computationally-efficient variant of our learning rule that is provably resilient to agents who do not behave as expected (as represented by a Byzantine adversary model) and deliberately try to spread misinformation.

5.1 Introduction

Given noisy data, the task of making meaningful inferences about a quantity of interest is at the heart of various complex estimation and detection problems arising in signal processing, information theory, machine learning, and control systems. When the information required to solve such problems is dispersed over a network, several interesting questions arise.

- How should the individual entities in the network combine their own private observations with the information received from neighbors to learn the quantity of interest?

- What are the minimal requirements on the information structure of the entities and the topology of the network for this to happen?

- How fast does information spread as a function of the diffusion rule and the structure of the network?

- What can be said when the underlying network changes with time and/or certain entities deviate from nominal behavior?

In this chapter, we provide rigorous theoretical answers to such questions for the setting where a group of agents receive a stream of private signals generated by an unknown quantity known as the "true state of the world". Communication among such agents is modeled by a graph. The goal of each agent is to eventually identify the true state from a finite set of hypotheses. However, while the *collective* signals across all agents might facilitate identification of the true state, signals received by any given agent may, in general, not be rich enough for identifying the state in isolation. Thus, the problem of interest is to develop and analyze local interaction rules that facilitate inference of the true state at every agent. The setup described above serves as a common mathematical abstraction for modeling and analyzing various decision-making problems in social and economic networks (e.g., opinion formation and spreading), and classification/detection problems arising in large-scale engineered systems (e.g., object recognition by a group of aerial robots).[1] While the former is typically studied under the moniker of non-Bayesian social learning, the latter usually goes by the name of distributed detection/hypothesis testing. In what follows, we discuss relevant literature.

[1] Although the model of interest to us (see Section 5.2) has been used to study decision-making in social networks [127–129], we do not claim that the rules developed in this chapter capture human reasoning in any way.

5.1.1 Related Work

Much of the earlier work on this topic of interest assumed the existence of a centralized fusion center for performing computational tasks [130–132]. Our work in this chapter, however, belongs to a more recent body of literature wherein individual agents are endowed with computational capabilities, and interactions among them are captured by a graph [127–129, 133–142]. These works are essentially inspired by the model in [127], where each agent maintains a belief vector (over the set of hypotheses) that is sequentially updated as the convex combination of its own Bayesian posterior and the priors of its neighbors. Subsequent approaches share a common theme: they typically involve a learning rule that combines a local Bayesian update with a consensus-based opinion pooling of neighboring beliefs. The key point of distinction among such rules stems from the specific manner in which neighboring opinions are aggregated. Specifically, linear opinion pooling is studied in [127, 128, 133, 134], whereas log-linear opinion pooling is studied in [135–142]. Under appropriate conditions on the observation model and the network structure, each of these approaches enable every agent to learn the true state exponentially fast, with probability 1. The rate of convergence, however, depends on the specific nature of the learning rule. Notably, finite-time concentration results are derived in [137–139], and a large-deviation analysis is conducted in [140, 141] for a broad class of distributions that generate the agents' observation profiles. Extensions to different types of time-varying graphs have also been considered in [133, 136–139]. In a recent paper [129], the authors go beyond specific functional forms of belief-update rules and, instead, adopt an axiomatic framework that identifies the fundamental factors responsible for social learning. We point out that belief-consensus algorithms on graphs have been studied prior to [127] as well as in [143, 144]. The model in [143, 144] differs from that in [127, 128, 133–142] in one key aspect: while in the former each agent has access to only one observation, the latter allows for influx of new information into the network in the form of a time-series of observations at every agent.

5.1.2 Summary of Contributions

In light of the above developments, we now elaborate on the main contributions of this work.

1) A Novel Distributed Learning Rule: In [138, Section III], the authors explain that the commonly studied linear and log-linear forms of belief aggregation are specific instances of a more general class of opinion pooling known as g-Quasi-Linear Opinion pools (g-QLOP), introduced in [145]. Our first contribution is the development of a novel belief update rule that deviates fundamentally from the broad family of g-QLOP learning rules. Specifically, the learning algorithm that we propose does not rely on any linear consensus-based belief aggregation protocol. Instead, each agent maintains two sets of belief vectors: a local belief vector and an actual belief vector. Each agent updates its local belief vector in a Bayesian manner based on only its private observations, i.e., without the influence of neighbors. The actual belief on each hypothesis is updated (up to normalization) as the *minimum* of the agent's own local belief and the actual beliefs of its neighbors on that particular hypothesis. We provide theoretical guarantees on the performance of this algorithm in Section 5.4. As we explain later in the chapter, establishing such guarantees requires proof techniques that differ substantially from those existing.

2) Strict Improvement in Rate of Learning: While data-aggregation via arithmetic or geometric averaging of neighboring beliefs allows asymptotic learning, such schemes may potentially dilute the rate at which false hypotheses are eliminated. In particular, for the linear consensus protocol introduced in [127], the limiting rate at which a particular false hypothesis is eliminated is almost surely upper-bounded by a quantity that depends on the relative entropies and centralities of the agents [128]. The log-linear rules in [137–141] improve upon such a rate: with probability 1, the asymptotic rate of rejection of a false hypothesis under such rules is a convex combination of the agents' relative entropies, where the convex weights correspond to the eigenvector centralities of the agents. In contrast, based on our approach, each false hypothesis is rejected by every agent exponentially fast, at a rate that is almost surely lower-bounded by the *best* relative entropy (between the true state and the false hypothesis) among all agents, provided the underlying network is static and strongly-connected. In Theorem 5.4.1, we show that the above result continues to hold even when the network changes with time, as long as a mild joint strong-connectivity condition is met. *Thus, to the best of our knowledge, our approach leads to a strict improvement in the rate of learning over all existing approaches: this constitutes our main contribution.*

3) Resilience to Adversaries: Despite the wealth of literature on distributed inference, there is limited understanding of the impact of misbehaving agents who do not follow the prescribed learning algorithm. Such agents may represent stubborn individuals or ideological extremists in the context of a social network, or model faults (either benign or malicious) in a networked control system. *In the presence of such misbehaving entities, how should the remaining agents process their private observations and the beliefs of their neighbors to eventually learn the truth?* To answer this question, we capture deviant behavior via the classical Byzantine adversary model [62], and develop a provably correct, resilient version of our proposed learning rule in Section 5.5. Theorem 5.5.1 characterizes the performance of this rule and, in particular, reveals that each regular agent can infer the truth exponentially fast. Furthermore, we identify conditions on the observation model and the network structure that guarantee applicability of our Byzantine-resilient learning rule, and argue that such conditions can be checked in polynomial time. The only related work that we are aware of in this regard is [142]. As we discuss in detail in Section 5.5, our proposed approach has various computational advantages relative to those in [142].

In addition to the main contributions discussed above, a minor contribution of this paper is the following. For static graphs where all agents behave normally, Corollary 5.4.3 establishes consistency of our learning rule under conditions that are necessary for *any* belief update rule to work, when agents make conditionally independent observations. In particular, we show that the typical assumption of strong-connectivity on the network can be relaxed, and identify the minimal requirement for uniquely learning any state that gets realized.[2] Despite its various advantages, our approach cannot, in general, handle the scenario where there does not exist any single true state that generates signals consistent with those seen by every agent. The method in [138, 139], however, is applicable to this case as well, and enables each agent to identify the hypothesis that *best* explains the groups' observations.

A preliminary version of the results in this chapter were published as [146]; these results were subsequently expanded upon in the pre-print [147].

5.2 Model and Problem Formulation

Network Model: We consider a group of agents $\mathcal{V} = \{1, 2, \ldots, n\}$ interacting over a time-varying, directed communication graph $\mathcal{G}[t] = (\mathcal{V}, \mathcal{E}[t])$, where $t \in \mathbb{N}$. An edge $(i, j) \in \mathcal{E}[t]$ indicates that agent i can directly transmit information to agent j at time-step t. If $(i, j) \in \mathcal{E}[t]$, then at time t, agent i will be called a neighbor of agent j, and agent j will be called an out-neighbor of agent i. The set $\mathcal{N}_i[t]$ will be used to denote the neighbors of agent i (excluding itself) at time t, whereas the set $\mathcal{N}_i[t] \cup \{i\}$ will be referred to as the inclusive neighborhood of agent i at time t. We will use $|\mathcal{C}|$ to denote the cardinality of a set $\mathcal{C}$.

Observation Model: Let $\Theta = \{\theta_1, \theta_2, \ldots, \theta_m\}$ denote m possible states of the world; each $\theta_i \in \Theta$ will be called a hypothesis. At each time-step $t \in \mathbb{N}_+$, every agent $i \in \mathcal{V}$ privately observes a signal $s_{i,t} \in \mathcal{S}_i$, where $\mathcal{S}_i$ denotes the signal space of agent i. The joint observation profile so generated across the network is denoted $s_t = (s_{1,t}, s_{2,t}, \ldots, s_{n,t})$, where $s_t \in \mathcal{S}$, and $\mathcal{S} = \mathcal{S}_1 \times \mathcal{S}_2 \times \ldots \mathcal{S}_n$. The signal s_t is generated based on a conditional likelihood function $l(\cdot|\theta^\star)$, governed by the true state of the world $\theta^\star \in \Theta$. Let $l_i(\cdot|\theta^\star), i \in \mathcal{V}$ denote the i-th marginal of $l(\cdot|\theta^\star)$. The

[2] A strongly-connected graph has a path between every pair of nodes.

signal structure of each agent $i \in \mathcal{V}$ is then characterized by a family of parameterized marginals $\{l_i(w_i|\theta) : \theta \in \Theta, w_i \in \mathcal{S}_i\}$.[3]

We make the following standard assumptions [127–129,133,134,136–142]: (i) The signal space of each agent i, namely $\mathcal{S}_i$, is finite. (ii) Each agent i has knowledge of its local likelihood functions $\{l_i(\cdot|\theta_p)\}_{p=1}^{m}$, and it holds that $l_i(w_i|\theta) > 0, \forall w_i \in \mathcal{S}_i$, and $\forall \theta \in \Theta$. (iii) The observation sequence of each agent is described by an i.i.d. random process over time; however, at any given time-step, the observations of different agents may potentially be correlated. (iv) There exists a fixed true state of the world $\theta^\star \in \Theta$ (unknown to the agents) that generates the observations of all the agents. Finally, we define a probability triple $(\Omega, \mathcal{F}, \mathbb{P}^{\theta^\star})$, where $\Omega \triangleq \{\omega : \omega = (s_1, s_2, \ldots), s_t \in \mathcal{S}, t \in \mathbb{N}_+\}$, $\mathcal{F}$ is the σ-algebra generated by the observation profiles, and $\mathbb{P}^{\theta^\star}$ is the probability measure induced by sample paths in Ω. Specifically, $\mathbb{P}^{\theta^\star} = \prod\limits_{t=1}^{\infty} l(\cdot|\theta^\star)$. For the sake of brevity, we will say that an event occurs almost surely to mean that it occurs almost surely w.r.t. the probability measure $\mathbb{P}^{\theta^\star}$.

Remark 5.2.1 *We point out that the existence of a true state that generates the private signals of all agents is a critical assumption for our approach to work; the method in [138] does not require this assumption. Moreover, unlike [135, 148], our rules do not apply to continuous parameter spaces.*

Note that assumptions (i) and (ii) on the observation model imply the existence of a constant $L \in (0, \infty)$ such that:

$$\max_{i \in \mathcal{V}} \max_{w_i \in \mathcal{S}_i} \max_{\theta_p, \theta_q \in \Theta} \left| \log \frac{l_i(w_i|\theta_p)}{l_i(w_i|\theta_q)} \right| \leq L. \tag{5.1}$$

We will make use of the above fact later in our analysis.

Given the above setup, the goal of each agent in the network is to discern the true state of the world $\theta^\star$. The challenge associated with such a task stems from the fact that the private signal structure of any given agent is in general only partially informative. To make this notion precise, define $\Theta_i^{\theta^\star} \triangleq \{\theta \in \Theta : l_i(w_i|\theta) = l_i(w_i|\theta^\star), \forall w_i \in$

[3] Whereas $w_i \in \mathcal{S}_i$ will be used to refer to a generic element of the signal space of agent i, $s_{i,t}$ will denote the random variable (with distribution $l_i(\cdot|\theta^\star)$) that corresponds to the observation of agent i at time-step t.

$\mathcal{S}_i\}$. In words, $\Theta_i^{\theta^\star}$ represents the set of hypotheses that are *observationally equivalent* to the true state $\theta^\star$ from the perspective of agent i. In general, for any agent $i \in \mathcal{V}$, we may have $|\Theta_i^{\theta^\star}| > 1$, necessitating collaboration among agents subject to the restrictions imposed by the time-varying communication topology.

Our **objective** in this chapter will be to design a distributed learning rule that allows each agent $i \in \mathcal{V}$ to identify the true state of the world asymptotically almost surely. To this end, we now introduce the following notion of source agents that will be useful in our subsequent developments.

Definition 5.2.1 *(**Source agents**) An agent i is said to be a source agent for a pair of distinct hypotheses $\theta_p, \theta_q \in \Theta$, if $K_i(\theta_p, \theta_q) > 0$, where $K_i(\theta_p, \theta_q)$ represents the KL-divergence between the distributions $l_i(\cdot|\theta_p)$ and $l_i(\cdot|\theta_q)$, and is given by:*[4]

$$K_i(\theta_p, \theta_q) = \sum_{w_i \in \mathcal{S}_i} l_i(w_i|\theta_p) \log \frac{l_i(w_i|\theta_p)}{l_i(w_i|\theta_q)}. \tag{5.2}$$

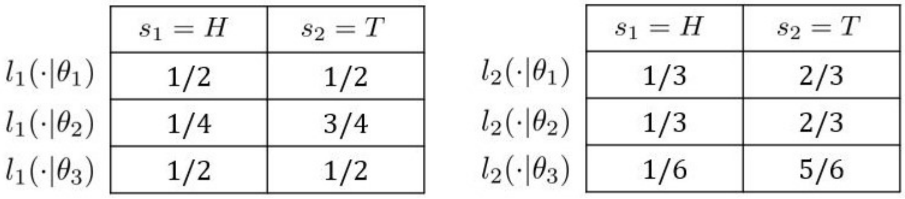
The set of all source agents for the pair θ_p, θ_q is denoted by $\mathcal{S}(\theta_p, \theta_q)$.

In words, a source agent for a pair $\theta_p, \theta_q \in \Theta$ is an agent that can distinguish between the pair of hypotheses θ_p, θ_q based on its private signal structure. It should be noted that $\mathcal{S}(\theta_p, \theta_q) = \mathcal{S}(\theta_q, \theta_p)$, since $K_i(\theta_p, \theta_q) > 0 \iff K_i(\theta_q, \theta_p) > 0$ [149]. In this work, we will assume that each state $\theta \in \Theta$ is *globally identifiable* w.r.t. the joint observation model of the entire network. Based on our terminology of source agents, this translates to the following.

Assumption 5.2.2 *(**Global Identifiability**) For each pair $\theta_p, \theta_q \in \Theta$ such that $\theta_p \neq \theta_q$, the set $\mathcal{S}(\theta_p, \theta_q)$ of agents that can distinguish between the pair θ_p, θ_q is non-empty.*

The above assumption is standard in the related literature. To illustrate the concepts described above, let us consider the following simple example.

[4] Although the standard notation for the KL-divergence between $l_i(\cdot|\theta_p)$ and $l_i(\cdot|\theta_q)$ is $D(l_i(\cdot|\theta_p)||l_i(\cdot|\theta_q))$, we use $K_i(\theta_p, \theta_q)$ as a shorthand for the same to avoid cluttering the exposition.



	$s_1 = H$	$s_2 = T$
$l_1(\cdot\|\theta_1)$	1/2	1/2
$l_1(\cdot\|\theta_2)$	1/4	3/4
$l_1(\cdot\|\theta_3)$	1/2	1/2

	$s_1 = H$	$s_2 = T$
$l_2(\cdot\|\theta_1)$	1/3	2/3
$l_2(\cdot\|\theta_2)$	1/3	2/3
$l_2(\cdot\|\theta_3)$	1/6	5/6

Fig. 5.1. Likelihood models for the two agents in Example 1. The model on the left is that of agent 1, while that on the right is of agent 2.

Example 1 *Consider a network of two agents with likelihood models as described in Fig. 5.1. At every time-step, each agent either observes heads H, or tails T. Thus, the common signal space for both agents is $\mathcal{S}_1 = \mathcal{S}_2 = \{H, T\}$. From Fig. 5.1, note that at each time-step, the probability of agent 1 observing H is 0.5 if either θ_1 or θ_3 gets realized, and 0.25 if θ_2 gets realized. Observe immediately that $\Theta_1^{\theta_1} = \{\theta_1, \theta_3\}$ and $\Theta_1^{\theta_2} = \{\theta_2\}$, i.e., agent 1 cannot distinguish between the states θ_1 and θ_3; however, it can tell θ_2 apart from either of the other two states. Agent 2's likelihood model can be interpreted similarly. Based on our terminology, we then have the following sets of source agents: $\mathcal{S}(\theta_1, \theta_2) = 1$, $\mathcal{S}(\theta_2, \theta_3) = \{1, 2\}$, and $\mathcal{S}(\theta_3, \theta_1) = 2$, implying global identifiability as per Assumption 5.2.2.*

In addition to Assumption 5.2.2, we will make a mild assumption on the time-varying communication topology. To this end, let the union graph over an interval $[t_1, t_2], 0 \leq t_1 < t_2$, indicate a graph with vertex set $\mathcal{V}$, and edge set $\bigcup_{\tau=t_1}^{t_2} \mathcal{E}[\tau]$. Based on this convention, we will assume (unless stated otherwise) that the sequence of communication graphs $\{\mathcal{G}[t]\}_{t=0}^{\infty}$ is *jointly strongly-connected*, in the following sense.

Assumption 5.2.3 ***(Joint Strong-Connectivity)*** *There exists $T \in \mathbb{N}_+$ such that the union graph over every interval of the form $[rT, (r+1)T)$ is strongly-connected, where $r \in \mathbb{N}$.*

While the above assumption on the network connectivity pattern is not necessary for solving the problem at hand, it is fairly standard in the analysis of distributed algorithms over time-varying networks [119, 120, 138]. Having introduced the model and the problem formulation, we now proceed to a formal description of our learning algorithm.

5.3 Proposed Learning Rule

In this section, we propose a novel belief update rule (Algorithm 3) and discuss the intuition behind it. Every agent i maintains and updates (at every time-step t) two separate sets of belief vectors, namely, $\boldsymbol{\pi}_{i,t}$ and $\boldsymbol{\mu}_{i,t}$. Each of these vectors are probability distributions over the hypothesis set Θ. We will refer to $\boldsymbol{\pi}_{i,t}$ and $\boldsymbol{\mu}_{i,t}$ as the "local" belief vector (for reasons that will soon become obvious), and the "actual" belief vector, respectively, maintained by agent i. The **goal** of each agent $i \in \mathcal{V}$ in the network will be to use its own private signals and the information available from its neighbors to update $\boldsymbol{\mu}_{i,t}$ sequentially, so that $\lim_{t\to\infty} \mu_{i,t}(\theta^*) = 1$ almost surely. To do so, at each time-step $t+1$ (where $t \in \mathbb{N}$), agent i does the following for each $\theta \in \Theta$. It first generates $\pi_{i,t+1}(\theta)$ via a local Bayesian update rule that incorporates the private observation $s_{i,t+1}$ using $\pi_{i,t}(\theta)$ as a prior (line 5 in Algo. 3). Having generated $\pi_{i,t+1}(\theta)$, agent i updates $\mu_{i,t+1}(\theta)$ (up to normalization) by setting it to be the *minimum* of its locally generated belief $\pi_{i,t+1}(\theta)$, and the actual beliefs $\mu_{j,t}(\theta), j \in \mathcal{N}_i[t] \cup \{i\}$ of its inclusive neighborhood at the previous time-step (line 6 in Algo. 3). It then reports $\boldsymbol{\mu}_{i,t+1}$ to each of its out-neighbors at time $t+1$.[5]

Intuition behind the learning rule: At the core of our learning algorithm are two key principles: (1) *Preservation of the intrinsic discriminatory capabilities of the agents*, and (2) *Propagation of low beliefs on each false hypothesis.* We now elaborate on these features.

Consider the set of source agents $\mathcal{S}(\theta^*, \theta)$ that can differentiate between a certain false hypothesis θ and the true state $\theta^\star$. By definition, the signal structures of such agents are rich enough for them to be able to eliminate θ on their own, i.e., without

[5] Note that based on our algorithm, agents only exchange their actual beliefs, and not their local beliefs.

Algorithm 3 Belief update rule for each $i \in \mathcal{V}$

```
Initialization: $\mu_{i,0}(\theta) > 0$, $\pi_{i,0}(\theta) > 0$, $\forall \theta \in \Theta$, and $\sum_{\theta \in \Theta} \mu_{i,0}(\theta) = 1$, $\sum_{\theta \in \Theta} \pi_{i,0}(\theta) = 1$
Transmit $\boldsymbol{\mu}_{i,0}$ to out-neighbors at time 0
for $t+1 \in \mathbb{N}_+$ do
    for $\theta \in \Theta$ do
        Update local belief on $\theta$ as
            $$\pi_{i,t+1}(\theta) = \frac{l_i(s_{i,t+1}|\theta)\pi_{i,t}(\theta)}{\sum_{p=1}^{m} l_i(s_{i,t+1}|\theta_p)\pi_{i,t}(\theta_p)} \tag{5.3}$$
        Update actual belief on $\theta$ as
            $$\mu_{i,t+1}(\theta) = \frac{\min\{\{\mu_{j,t}(\theta)\}_{j \in \mathcal{N}_i[t] \cup \{i\}}, \pi_{i,t+1}(\theta)\}}{\sum_{p=1}^{m} \min\{\{\mu_{j,t}(\theta_p)\}_{j \in \mathcal{N}_i[t] \cup \{i\}}, \pi_{i,t+1}(\theta_p)\}} \tag{5.4}$$
    end for
    Transmit $\boldsymbol{\mu}_{i,t+1}$ to out-neighbors at time $t+1$
end for
```

the support of their neighbors. To achieve this, we require each agent to maintain a local belief vector that is updated (via (5.3)) *without any network influence* using only the agent's own private signals. Doing so ensures that $\pi_{i,t}(\theta) \to 0$ a.s. for each $i \in \mathcal{S}(\theta^\star, \theta)$. Next, leveraging this property, we want to be able to propagate low beliefs on θ from $\mathcal{S}(\theta^\star, \theta)$ to $\mathcal{V} \setminus \mathcal{S}(\theta^\star, \theta)$, i.e., the agents in $\mathcal{S}(\theta^*, \theta)$ should contribute towards driving the actual beliefs of their out-neighbors (and eventually, of all the agents in the set $\mathcal{V} \setminus \mathcal{S}(\theta^\star, \theta)$) on the hypothesis θ to zero. Using a min-rule of the form (5.4), with $\pi_{i,t+1}(\theta)$ featuring as an external network-independent input, facilitates such propagation without compromising the abilities of agents in $\mathcal{S}(\theta^\star, \theta)$ to eliminate θ. When set in motion, our learning rule triggers a process of belief reduction on θ originating at $\mathcal{S}(\theta^\star, \theta)$ that eventually propagates to each agent in the network reachable from $\mathcal{S}(\theta^\star, \theta)$.

Remark 5.3.1 *We emphasize that the proposed learning rule given by Algorithm 3 does not employ any form of "belief-averaging". This feature is in stark contrast with existing approaches to distributed hypothesis testing that rely either on linear opinion pooling [127, 128, 133, 134], or log-linear opinion pooling [135–142]. As such, the lack of linearity in our belief update rule precludes (direct or indirect) adaptation of existing analysis techniques to suit our needs.*

5.4 Analysis of Algorithm 3

5.4.1 Statement of the Results

In this section, we characterize the performance of Algorithm 3. We start with one of the main results of the paper, proven in Section 5.8.1. Before stating the result, we remind the reader that for an agent i, $K_i(\theta_p, \theta_q)$ represents the KL-divergence between the distributions $l_i(\cdot|\theta_p)$ and $l_i(\cdot|\theta_q)$, and captures agent i's ability to distinguish between the states θ_p and θ_q.

Theorem 5.4.1 *Suppose the observation model satisfies the global identifiability condition (Assumption 5.2.2), and the sequence of communication graphs $\{\mathcal{G}[t]\}_{t=0}^{\infty}$ is jointly strongly-connected (Assumption 5.2.3). Then, Algorithm 3 provides the following guarantees.*

- ***(Consistency)**: For each agent $i \in \mathcal{V}$, $\mu_{i,t}(\theta^\star) \rightarrow 1$ a.s.*
- ***(Asymptotic Rate of Rejection of False Hypotheses)**: Consider any false hypothesis $\theta \in \Theta \setminus \{\theta^\star\}$. Then, the following holds for each agent $i \in \mathcal{V}$:*

$$\liminf_{t\to\infty} -\frac{\log \mu_{i,t}(\theta)}{t} \geq \max_{v\in\mathcal{S}(\theta^\star,\theta)} K_v(\theta^\star, \theta) \ a.s. \tag{5.5}$$

The above result tells us that with probability 1, every agent i will be able to rule out each false hypothesis θ exponentially fast, at a rate that is eventually lower-bounded by the *best* KL-divergence across the network between the pair of hypotheses

$\theta^\star$ and θ. In particular, this implies that given any $\epsilon > 0$, the probability that agent i's instantaneous rate of rejection of θ, namely $-\log \mu_{i,t}(\theta)/t$, is lower than the quantity $\max_{v\in\mathcal{S}(\theta^\star,\theta)} K_v(\theta^\star,\theta)$ by an additive factor of ϵ, decays to zero. The next result, proven in Section 5.8.2, sheds some light on the rate of decay of this probability.

Theorem 5.4.2 *Suppose the conditions in Theorem 5.4.1 hold. Fix $\theta \in \Theta \setminus \{\theta^\star\}$, and let $\bar{K}(\theta^\star,\theta) = \max_{v\in\mathcal{S}(\theta^\star,\theta)} K_v(\theta^\star,\theta)$. Then for every $\epsilon > 0$ and $\delta \in (0,1)$, there exists a set $\Omega'(\delta) \subseteq \Omega$ with $\mathbb{P}^{\theta^\star}(\Omega'(\delta)) \geq 1-\delta$, such that the following holds for each agent $i \in \mathcal{V}$:*

$$\liminf_{t\to\infty} -\frac{1}{t}\log \mathbb{P}^{\theta^\star}\left(\left\{-\frac{\log \mu_{i,t}(\theta)}{t} \leq \bar{K}(\theta^\star,\theta) - \epsilon\right\} \cap \Omega'(\delta)\right) \geq \frac{\epsilon^2}{8L^2}. \tag{5.6}$$

Our next result pertains to the special case when the communication graph does not change over time, i.e., when $\mathcal{G}[t] = \mathcal{G}, \forall t \in \mathbb{N}$. To state the result, we will employ the following terminology. Given two disjoint sets $\mathcal{C}_1, \mathcal{C}_2 \subseteq \mathcal{V}$, we say $\mathcal{C}_2$ is reachable from $\mathcal{C}_1$ if for every $i \in \mathcal{C}_2$, there exists a directed path in $\mathcal{G}$ from some $j \in \mathcal{C}_1$ to agent i (note that j will in general be a function of i).

Corollary 5.4.3 *Let the communication graph be time-invariant and be denoted by $\mathcal{G}$. Suppose the following conditions hold. (i) The observation model satisfies the global identifiability condition (Assumption 5.2.2). (ii) For every pair of hypotheses $\theta_p \neq \theta_q \in \Theta$, the set $\mathcal{V} \setminus \mathcal{S}(\theta_p,\theta_q)$ is reachable from the set $\mathcal{S}(\theta_p,\theta_q)$ in $\mathcal{G}$. Then, Algorithm 3 guarantees consistency as in Theorem 5.4.1. Furthermore, for every $\theta \in \Theta \setminus \{\theta^\star\}$, the following holds for each agent $i \in \mathcal{V}$:*

$$\liminf_{t\to\infty} -\frac{\log \mu_{i,t}(\theta)}{t} \geq \max_{v\in\mathcal{S}_i(\theta^\star,\theta)} K_v(\theta^\star,\theta) \; a.s., \tag{5.7}$$

where $\mathcal{S}_i(\theta^\star,\theta) \subseteq \mathcal{S}(\theta^\star,\theta)$ are those source agents from which there exists a directed path to i in $\mathcal{G}$.

Proof Fix $\theta \in \Theta \setminus \{\theta^\star\}$, and consider an agent $i \in \mathcal{V} \setminus \mathcal{S}(\theta^\star,\theta)$. The sets $\mathcal{S}(\theta^\star,\theta)$ and $\mathcal{S}_i(\theta^\star,\theta)$ are non-empty based on conditions (i) and (ii) of the theorem, respectively.

Following a similar line of argument as in the proof of Theorem 5.4.1, one can establish the following for each $v \in \mathcal{S}_i(\theta^\star, \theta)$.

$$\liminf_{t\to\infty} -\frac{\log \mu_{i,t}(\theta)}{t} \geq K_v(\theta^\star, \theta) \ a.s. \tag{5.8}$$

The assertion regarding equation (5.7) then follows readily. Consistency follows by noting that since $\mathcal{S}_i(\theta^\star, \theta) \subseteq \mathcal{S}(\theta^\star, \theta)$, $K_v(\theta^\star, \theta) > 0, \forall v \in \mathcal{S}_i(\theta^\star, \theta)$. ■

Our next result reveals that the combination of conditions (i) and (ii) in Corollary 5.4.3 constitutes *minimal* requirements on the observation model and the network structure for *any* learning algorithm to guarantee consistency, when the observations of the agents are conditionally independent.

Theorem 5.4.4 *Let the communication graph be time-invariant and be denoted by $\mathcal{G}$. Then, the following assertions hold.*

(i) Conditions (i) and (ii) in Corollary 5.4.3, taken together, is equivalent to global identifiability of each source component of $\mathcal{G}$.[6]

(ii) Suppose the observations of the agents are independent conditional on the realization of any state, i.e., $l(\cdot|\theta) = \prod_{i=1}^{n} l_i(\cdot|\theta), \forall \theta \in \Theta$. Then, global identifiability of each source component of $\mathcal{G}$ is necessary and sufficient for unique identification of any true state that gets realized, at every agent, with probability 1.

The proof of the above result is fairly straightforward and hence omitted here. We now leverage the above results to quantify the rate at which the overall network uncertainty about the true state decays to zero. To measure such uncertainty, we employ the following metric from [128] which captures the total variation distance between the agents' beliefs at time-step t, and the probability distribution that is concentrated entirely on the true state of the world, namely $\mathbf{1}_{\theta^\star}(\cdot)$:

$$e_t(\theta^\star) \triangleq \frac{1}{2}\sum_{i=1}^{n} \left\|\boldsymbol{\mu}_{i,t}(\cdot) - \mathbf{1}_{\theta^\star}(\cdot)\right\|_1 = \sum_{i=1}^{n}\sum_{\theta\neq\theta^\star} \mu_{i,t}(\theta). \tag{5.9}$$

[6] A source component of a time-invariant graph $\mathcal{G}$ is a strongly connected component with no incoming edges.

Given that $\theta^\star$ gets realized, the *rate of social learning* is then defined as [128, 140]:

$$\rho_L(\theta^\star) \triangleq \liminf_{t\to\infty} -\frac{1}{t}\log e_t(\theta^\star). \tag{5.10}$$

Notice that the above expression depends on the state being realized; to account for the realization of any state, one can simply look at the quantity $\min_{\theta^\star\in\Theta}\rho_L(\theta^\star)$ that provides a sense for the least rate of learning one can expect given a certain observation model, a network, and a consistent learning algorithm. We have the following simple results; their proofs are trivial and hence omitted.

Corollary 5.4.5 *Suppose the conditions stated in Theorem 5.4.1 are met. Then, Algorithm 3 guarantees:*

$$\rho_L(\theta^\star) \geq \min_{\theta\neq\theta^\star}\max_{v\in\mathcal{S}(\theta^\star,\theta)} K_v(\theta^\star,\theta)\ a.s. \tag{5.11}$$

Corollary 5.4.6 *Suppose the conditions stated in Corollary 5.4.3 are met. Then, Algorithm 3 guarantees:*

$$\rho_L(\theta^\star) \geq \min_{\theta\neq\theta^\star}\min_{i\in\mathcal{V}}\max_{v\in\mathcal{S}_i(\theta^\star,\theta)} K_v(\theta^\star,\theta)\ a.s. \tag{5.12}$$

5.4.2 Discussion of the Results

Comments on Theorem 5.4.1: Let us compare the rate of learning based on our method to those existing in literature. Under identical assumptions of global identifiability of the observation model, and strong-connectivity (or joint strong-connectivity as in [138]) of the underlying communication graph, both linear [127, 128] and log-linear [137, 138, 140] opinion pooling lead to an asymptotic rate of rejection of the form $\sum_{i\in\mathcal{V}}\nu_i K_i(\theta^\star,\theta)$ for each false hypothesis $\theta\in\Theta\setminus\{\theta^\star\}$, for each agent $i\in\mathcal{V}$.[7]

[7] In [138], the consensus weights are chosen to obtain a network-structure independent (albeit network-size dependent) rate of rejection of θ of the form $1/n\sum_{i\in\mathcal{V}}K_i(\theta^\star,\theta)$. The same rate is obtained with static, undirected networks when the consensus weight matrix is symmetric, since the eigenvector centralities are simply $1/n$ in such a case.

Here, ν_i represents the eigenvector centrality of agent $i \in \mathcal{V}$, which is strictly positive for a strongly-connected graph. Thus, referring to equation (5.5) reveals that the asymptotic rate of rejection of each false hypothesis (and hence, the rate of social learning) resulting from our algorithm (see (5.11)), is a *strict improvement* over all existing rates - this constitutes a significant contribution of our work. Furthermore, observe from Corollary 5.4.5 that the lower bound on the rate of social learning is *independent of both the size and structure of the network.* A key implication of this result is the fact that as long as the total information content of the network remains the same, the specific manner in which signals are allocated to agents does not impact the long-run learning rate of our approach. In sharp contrast, existing learning rates that depend on the agents' eigenvector centralities may suffer under poor signal allocations; see [128] for a discussion on this topic.

It should, however, be noted that the network independence aspect of our approach concerns *asymptotic* learning rates. The dependence on the network structure (presumably, on the diameter) is bound to manifest itself in the transients generated by our rule. Given the non-linear structure of our update rule (5.4), characterizing such a dependence is quite non-trivial.

Comments on Theorem 5.4.2: At any given time t, for some $i \in \mathcal{V}$ and $\theta \neq \theta^\star$, let us consider the set of all sample paths where agent i's instantaneous rate of rejection of θ is lower than its asymptotic lower bound by a constant additive factor of ϵ. Theorem 5.4.2 complements Theorem 5.4.1 by telling us that an arbitrarily accurate approximation of the measure of such "bad" sample paths eventually decays to zero at an exponential rate no smaller than $\epsilon^2/8L^2$ (the approximation is arbitrarily accurate since the set $\Omega'(\delta)$ can be chosen to have measure arbitrarily close to 1). It is instructive to compare the concentration result of Theorem 5.4.2 with [138, Theorem 2], [140, Theorem 2], and [137, Lemma 3]. The analogous results in these papers are more elegant relative to ours, since they do not involve a set of the form $\Omega'(\delta)$ that shows up in our analysis. A refinement of Theorem 5.4.2 to obtain a cleaner non-asymptotic result would require a precise characterization of the transient dynamics generated by our learning rule: we reserve investigations along this line as future work.

Comments on Corollary 5.4.3: While Theorem 5.4.4 identifies an *algorithm-independent* necessary condition for ensuring unique identifiability of any realized state at every agent (when the communication graph is time-invariant and agents receive conditionally independent signals), Corollary 5.4.3 reveals that such a condition is also sufficient for our proposed learning algorithm to work. We believe that a result of this flavor is missing in the existing literature on distributed hypothesis testing, where strong-connectivity is a standard assumption. The authors in [150] do relax the strong-connectivity assumption, but require *every* strongly-connected component of $\mathcal{G}$ to be globally identifiable for learning to take place [150, Proposition 4]. In contrast, Corollary 5.4.3 requires only the source components of $\mathcal{G}$ to satisfy the global identifiability requirement. Interestingly, our conclusions in this context align with an analogous result that identifies joint detectability of each source component as the minimal requirement for solving the related problem of distributed state estimation [1, 58]. The more general network condition in Corollary 5.4.3 (as opposed to strong-connectivity) comes at the cost of a potential reduction in the rate of social learning, as reflected in Corollary 5.4.6. When the underlying graph is strongly-connected, $\mathcal{S}_i(\theta^\star, \theta) = \mathcal{S}(\theta^\star, \theta)$. Consequently, the min w.r.t. the agent set $\mathcal{V}$ in equation (5.12) goes away, and we recover Corollary 5.4.5.

5.5 Learning despite Misinformation

In this section, we will address the problem of learning the true state of the world despite the presence of certain agents who do not behave as expected and deliberately try to spread misinformation. In order to isolate the challenges introduced by such malicious entities, we will consider a time-invariant communication graph $\mathcal{G}$ for our subsequent discussion; we anticipate that our proposed approach will extend to the time-varying case with suitable modifications.[8]

Adversary Model: As our adversary model, we consider the same worst-case Byzantine attack model that was discussed in Chapter 3. Recall that Byzantine agents possess complete knowledge of the observation model, the network model, the algorithms being used, the information being exchanged, and the true state of the world. In return for endowing the adversaries with such capabilities, we will consider an f-local adversarial model, i.e., we assume that there are at most f adversaries in the neighborhood of any non-adversarial agent, where $f \in \mathbb{N}$. As in Chapter 3, the adversarial set will be denoted by $\mathcal{A} \subset \mathcal{V}$, and the remaining agents $\mathcal{R} = \mathcal{V} \setminus \mathcal{A}$ will be called the regular agents.

Our immediate goals are as follows. (i) Devise an algorithm that enables each regular agent to asymptotically identify the true state with probability 1, despite the presence of an f-local Byzantine adversarial set. (ii) Identify conditions on the observation model and the network structure that guarantee correctness of such an algorithm. Prior to addressing these goals, we briefly motivate the need for a novel Byzantine-resilient learning algorithm.

Motivation: A standard way to analyze the impact of adversarial agents while designing resilient distributed consensus-based protocols (for applications in consensus [87, 88], optimization [89, 90], hypothesis testing [142], and multi-agent rendezvous [95]) is to construct an equivalent matrix representation of the linear update rule that involves only the regular agents [152]. In particular, this requires expressing the iterates of a regular agent as a convex combination of the iterates of its regular neighbors, based on appropriate filtering techniques, and under certain assumptions on the network structure. While this can indeed be achieved efficiently for scalar consensus problems, for problems requiring consensus on vectors (like the belief vectors in our setting), such an approach typically requires the computation of sets known

[8] Different from our setting, the *forceful* agents in [151] do not behave arbitrarily and, in fact, update their beliefs (even if infrequently) by interacting with their neighbors; our adversary model makes no such assumptions.

as *Tverberg partitions*. However, there is no known algorithm that can compute an exact Tverberg partition in polynomial time for a general d-dimensional finite point set [153]. Consequently, since the filtering approach developed in [142] requires each regular agent to compute a Tverberg partition at every iteration, the resulting computations are forbiddingly high. The authors in [142] do briefly discuss an alternate pairwise learning rule that requires agents to perform scalar consensus on relative confidence levels (instead of beliefs) of one hypothesis over another. Under such a rule, for each regular agent, its relative confidence on the true state over every false hypothesis approaches infinity - a condition that is difficult to verify in practice. Moreover, the pairwise learning rule in [142] requires each agent to maintain and update at each time-step a vector of dimension $O(m^2)$. In contrast, we propose a simple, light-weight Byzantine-resilient learning rule that avoids the computation of Tverberg partitions, and requires agents to update two m-dimensional belief vectors.

5.5.1 A Byzantine-Resilient Distributed Learning Rule

In this section, we develop an easy to implement and computationally-efficient extension of Algorithm 3 that guarantees learning despite the presence of Byzantine adversaries. We call it the Local-Filtering based Resilient Hypothesis Elimination (LFRHE) algorithm (Algorithm 4). Like Algorithm 3, the LFRHE algorithm requires every regular agent i to maintain and update (at every time-step t) a local belief vector $\boldsymbol{\pi}_{i,t}$, and an actual belief vector $\boldsymbol{\mu}_{i,t}$. While $\boldsymbol{\pi}_{i,t}$ is updated as before via (5.3), the update of $\boldsymbol{\mu}_{i,t}$ is the key feature of Algorithm 4. To update $\mu_{i,t+1}(\theta)$, agent $i \in \mathcal{R}$ first checks whether it has at least $2f+1$ neighbors. If it does, then it rejects the highest f and the lowest f neighboring beliefs $\mu_{j,t}(\theta), j \in \mathcal{N}_i$ (line 7 in Algo. 4), and employs a min-rule as before, but using only the remaining beliefs (line 8 in Algo. 4). Thus, agent i filters out the most extreme neighboring beliefs on each hypothesis, and retains only the moderate ones to update its own actual belief. If agent i has strictly

Algorithm 4 Belief update rule for each $i \in \mathcal{R}$

```
1: Initialization: $\mu_{i,0}(\theta) > 0$, $\pi_{i,0}(\theta) > 0$, $\forall \theta \in \Theta$, and $\sum_{\theta\in\Theta} \mu_{i,0}(\theta) = 1$, $\sum_{\theta\in\Theta} \pi_{i,0}(\theta) = 1$
2: Transmit $\boldsymbol{\mu}_{i,0}$ to out-neighbors
3: for $t+1 \in \mathbb{N}_+$ do
4:     for $\theta \in \Theta$ do
5:         Update local belief on $\theta$ as per (5.3)
6:         if $|\mathcal{N}_i| \geq (2f+1)$ then
7:             Sort $\mu_{j,t}(\theta), j \in \mathcal{N}_i$ from highest to lowest, and reject the highest $f$ and the lowest $f$ of such beliefs.
8:             Let $\mathcal{M}^{\theta}_{i,t}$ be the set of agents whose beliefs are not rejected in the previous step. Update $\mu_{i,t+1}(\theta)$ as
                   $$\mu_{i,t+1}(\theta) = \frac{\min\{\{\mu_{j,t}(\theta)\}_{j\in\mathcal{M}^{\theta}_{i,t}}, \pi_{i,t+1}(\theta)\}}{\sum\limits_{p=1}^{m} \min\{\{\mu_{j,t}(\theta_p)\}_{j\in\mathcal{M}^{\theta_p}_{i,t}}, \pi_{i,t+1}(\theta_p)\}} \quad (5.13)$$
9:         else
10:            Update $\mu_{i,t+1}(\theta)$ as
                   $$\mu_{i,t+1}(\theta) = \pi_{i,t+1}(\theta) \quad (5.14)$$
11:        end if
12:    end for
13:    Transmit $\boldsymbol{\mu}_{i,t+1}$ to out-neighbors
14: end for
```

fewer than $2f+1$ neighbors, then it decides against using neighboring information and, instead, updates its actual belief vector to be equal to its local belief vector (line 10 in Algo. 4).

To state our main result concerning the correctness of Algorithm 4, we recall the following definitions from Chapter 3.

Definition 5.5.1 *(r-**reachable set**) [88] For a graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$, a set $\mathcal{C} \subseteq \mathcal{V}$, and an integer $r \in \mathbb{N}_+$, $\mathcal{C}$ is an r-reachable set if there exists an $i \in \mathcal{C}$ such that $|\mathcal{N}_i \backslash \mathcal{C}| \geq r$.*

Definition 5.5.2 *(**strongly** r-**robust graph** w.r.t. $\mathcal{S}(\theta_p, \theta_q)$) For $r \in \mathbb{N}_+$ and $\theta_p, \theta_q \in \Theta$, a graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ is strongly r-robust w.r.t. the set of source agents $\mathcal{S}(\theta_p, \theta_q)$, if for every non-empty subset $\mathcal{C} \subseteq \mathcal{V} \setminus \mathcal{S}(\theta_p, \theta_q)$, $\mathcal{C}$ is r-reachable.*

Theorem 5.5.1 *Suppose that for every pair of hypotheses $\theta_p, \theta_q \in \Theta$, the graph $\mathcal{G}$ is strongly $(2f+1)$-robust w.r.t. the source set $\mathcal{S}(\theta_p, \theta_q)$. Then, Algorithm 4 guarantees the following despite the actions of any f-local set of Byzantine adversaries.*

- ***(Consistency)**: For each agent $i \in \mathcal{R}$, $\mu_{i,t}(\theta^\star) \to 1$ a.s.*
- ***(Asymptotic Rate of Rejection of False Hypotheses)**: Consider any false hypothesis $\theta \in \Theta \setminus \{\theta^\star\}$. Then, the following holds for each agent $i \in \mathcal{R}$.*

$$\liminf_{t \to \infty} -\frac{\log \mu_{i,t}(\theta)}{t} \geq \min_{v \in \mathcal{S}(\theta^\star, \theta) \cap \mathcal{R}} K_v(\theta^\star, \theta) \text{ a.s.} \tag{5.15}$$

Proof See Section 5.8.3. ■

Remark 5.5.2 *For any pair $\theta_p, \theta_q \in \Theta$, notice that the strong-robustness condition in Theorem 5.5.1 (together with Def. 3.7.2) requires $|\mathcal{S}(\theta_p, \theta_q)| \geq (2f+1)$, if $\mathcal{V} \setminus \mathcal{S}(\theta_p, \theta_q)$ is non-empty. In particular, it blends requirements on the signal structures of the agents with those on the communication graph. To gain intuition about this condition, suppose $\Theta = \{\theta_1, \theta_2\}$, and consider an agent $i \in \mathcal{V} \setminus \mathcal{S}(\theta_1, \theta_2)$. To enable i to learn the truth despite potential adversaries in its neighborhood, one requires (i) redundancy in the signal structures of the agents, and (ii) redundancy in the network structure to ensure reliable information flow from $\mathcal{S}(\theta_1, \theta_2)$ to agent i. These requirements are encapsulated by Theorem 5.5.1. For a fixed source set $\mathcal{S}(\theta_p, \theta_q)$, checking whether $\mathcal{G}$ is strongly $(2f+1)$-robust w.r.t. $\mathcal{S}(\theta_p, \theta_q)$ can be done in polynomial time by drawing connections to the process of bootstrap percolation on networks [85, Proposition 5]. Since the source sets for each pair $\theta_p, \theta_q \in \Theta$ can also be computed in polynomial* *time via a simple inspection of the agents' signal structures, it follows that the strong-robustness condition in Theorem 5.5.1 can be checked in polynomial time.*

Leveraging Theorem 5.5.1, we can characterize the rate of decay of the collective uncertainty of the regular agents regarding the true state. To do so, we employ the following modification of the metric (5.9):

$$e_t^{\mathcal{R}}(\theta^\star) \triangleq \frac{1}{2}\sum_{i\in\mathcal{R}} \left\|\boldsymbol{\mu}_{i,t}(\cdot) - \mathbf{1}_{\theta^\star}(\cdot)\right\|_1 = \sum_{i\in\mathcal{R}}\sum_{\theta\neq\theta^\star} \mu_{i,t}(\theta). \tag{5.16}$$

Note that this metric only considers the beliefs of the regular agents as the Byzantine agents can update their beliefs however they wish. With $\theta^\star$ as the true state, we define the rate of social learning in the presence of Byzantine adversaries as:

$$\rho_L^{\mathcal{R}}(\theta^\star) \triangleq \liminf_{t\to\infty} -\frac{1}{t}\log e_t^{\mathcal{R}}(\theta^\star). \tag{5.17}$$

We have the following immediate corollary of Theorem 5.5.1.

Corollary 5.5.3 *Suppose the conditions stated in Theorem 5.5.1 are met. Then, Algorithm 4 guarantees:*

$$\rho_L^{\mathcal{R}}(\theta^\star) \geq \min_{\theta\neq\theta^\star}\ \min_{v\in\mathcal{S}(\theta^\star,\theta)\cap\mathcal{R}} K_v(\theta^\star,\theta)\ a.s. \tag{5.18}$$

5.6 Simulations

Example 1 (Impact of Network Size on Rate of Convergence): For our first simulation study, we consider a binary hypothesis testing problem, i.e., $\Theta = \{\theta_1, \theta_2\}$, where the signal space for each agent is identical and comprises of signals w_1 and w_2. The (time-invariant) undirected network for this example is depicted in Figure 5.2(a). The likelihood models of the agents are as follows: $l_1(w_1|\theta_1) = 0.7, l_1(w_1|\theta_2) = 0.5$, and $l_i(w_1|\theta_1) = l_i(w_1|\theta_2) = 0.5, \forall i \in \mathcal{V}\setminus\{1\}$, i.e., agent 1 is the only informative agent. In order to compare the performance of Algorithm 3 to the linear and log-linear belief update rules in [127] and [138], we implement the latter assuming consensus weights are assigned based on the lazy Metropolis scheme (see [138] for details). Based on

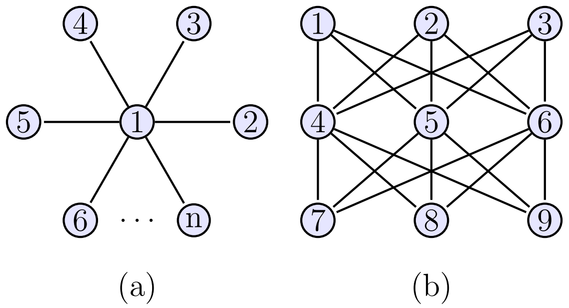


Fig. 5.2. Figures 5.2(a) and 5.2(b) represent the network models for simulation examples 1 and 2, respectively.

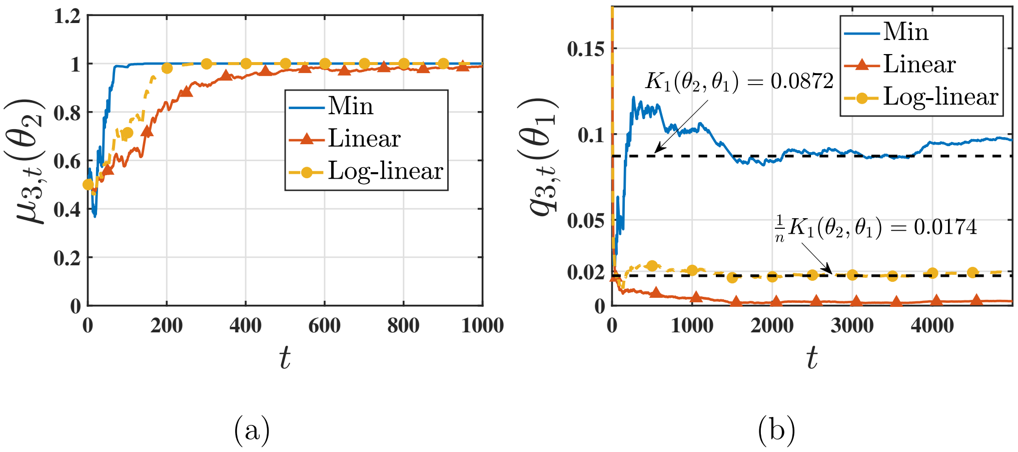


Fig. 5.3. Consider the setup of simulation example 1 with $n = 5$ agents. Fig. 5.3(a) depicts the evolution of agent 3's belief on the true state θ_2, and Fig. 5.3(b) depicts the evolution of the instantaneous rate of rejection of θ_1 for agent 3, namely $q_{3,t}(\theta_1) = -\log \mu_{3,t}(\theta_1)/t$.

this weight assignment, it is easy to verify that the eigenvector centrality of each agent is $1/n$. All agents start out with uniform priors. With $\theta^\star = \theta_2$, and $n = 5$, Figure 5.3 illustrates the performance of the three algorithms w.r.t. agent 3. In particular, Figure 5.3(a) reveals that based on our approach, agent 3's belief on the true state θ_2 converges to 1 faster than the other algorithms. Figure 5.3(b) makes this

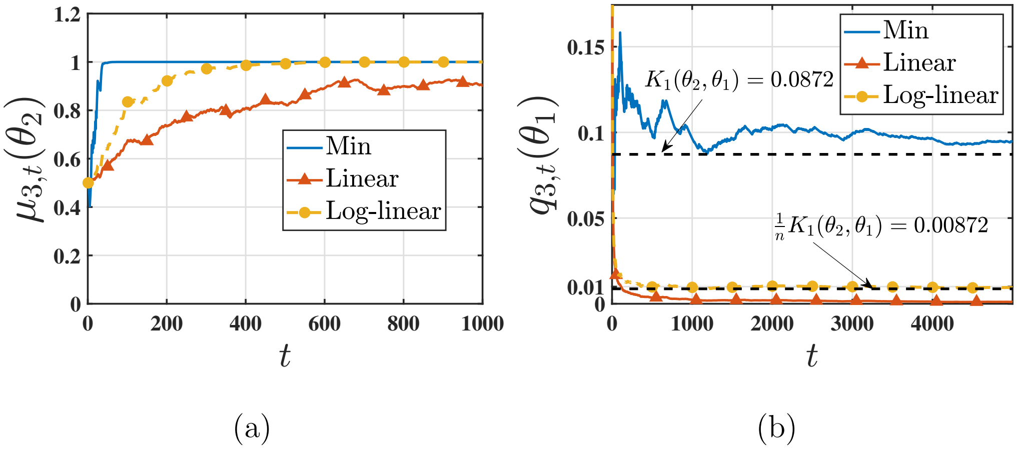


Fig. 5.4. Consider the setup of simulation example 1 with $n = 10$ agents. Fig. 5.4 illustrates the dilution in the rates of social learning for the linear and log-linear rules with an increase in the number of uninformative agents. Figures 5.4(a) and 5.4(b) are analogous to those in Figure 5.3.

observation precise by plotting the instantaneous rate of rejection of θ_1 for agent 3, namely $q_{3,t}(\theta_1) = -\log \mu_{3,t}(\theta_1)/t$. Consistent with the respective theoretical findings, $q_{3,t}(\theta_1)$ is eventually lower-bounded by $K_1(\theta_2,\theta_1)$ for our algorithm (see Theorem 5.4.1), approaches $K_1(\theta_2,\theta_1)/n$ for the log-linear rule in [138], and is eventually upper-bounded by $K_1(\theta_2,\theta_1)/n$ for the linear rule in [127]. Similar conclusions hold for the other agents.

Suppose we now double the number of agents in the network. Agent 1 continues to remain the only informative agent. Figure 5.4 compares the performances of the three algorithms for this case. Notably, the convergence rate for our approach remains unaffected, whereas that for the linear and log-linear rules gets diluted. This observation can be attributed to the fact that while the rate provided by our algorithm is both network-structure and network-size independent for strongly-connected networks (see Section 5.4.2), the rates of the linear and log-linear rules depend crucially on the eigenvector centralities of the agents, which, in this case, correspond to $1/n$. Thus, the gap between the performance of our algorithm, and that of the

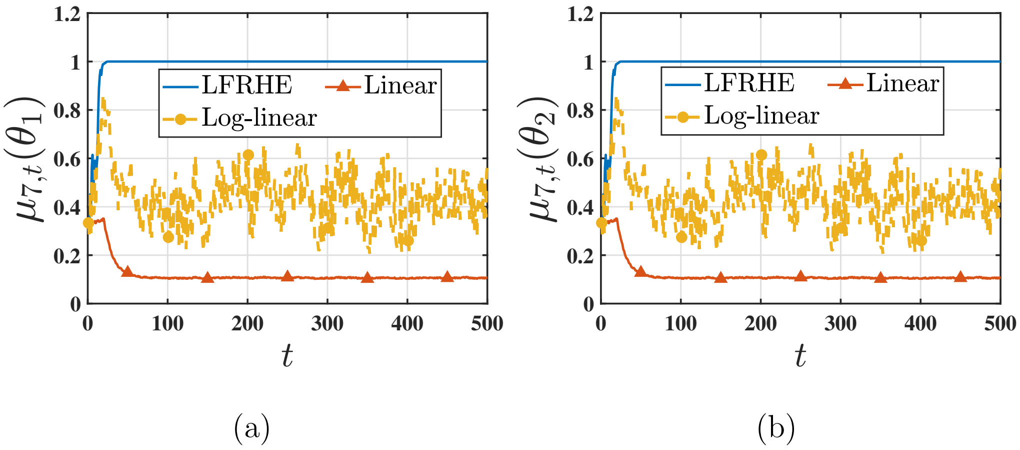


Fig. 5.5. Consider the setup of simulation example 2, where agent 5 acts as an adversary. Figures 5.5(a) and 5.5(b) depict the evolution of agent 7's belief on the true state, when $\theta^\star = \theta_1$, and $\theta^\star = \theta_2$, respectively.

linear and log-linear update rules (as measured by convergence rates), becomes more pronounced as the number of uninformative agents increase (i.e., as n increases, but the total information content of the network remains the same).

Example 2 (Impact of Adversaries): While the previous example highlighted the benefits of Algorithm 3, we now focus on an example that demonstrates the resilience of its variant, namely the LFRHE algorithm (Algorithm 4), to the presence of Byzantine adversaries. To this end, consider the undirected network in Figure 5.2(b). For this example, $\Theta = \{\theta_1, \theta_2, \theta_3\}$, and $\mathcal{S}_i = \{w_1, w_2\}, \forall i \in \mathcal{V}$. Suppose the agent likelihood models are given by $l_i(w_1|\theta_1) = 3/4, l_i(w_1|\theta_2) = l_i(w_1|\theta_3) = 1/3, \forall i \in \{1,2,3\}$, $l_i(w_1|\theta_1) = l_i(w_1|\theta_2) = 2/5, l_i(w_1|\theta_3) = 1/7, \forall i \in \{4,5,6\}$, and $l_i(w_1|\theta_1) = l_i(w_1|\theta_2) = 1/2, l_i(w_1|\theta_3) = 5/6, \forall i \in \{7,8,9\}$. Suppose $f = 1$ and agent 5 is the only adversarial agent. It is easy to see that condition (i) in Theorem 5.5.1 is met. We will compare the performance of Algorithm 4 with the linear rule in [127], and the log-linear rule in [138]. For implementing the latter, we again assign consensus weights based on the lazy Metropolis scheme. All agents start out with uniform priors. The adversary, agent 5, maintains a belief of 0.1 on the true state, and 0.45 on each of the false hypotheses, for all $t \geq 20$. Figures 5.5(a) and 5.5(b) illustrate the repercussions of this action on agent 7, when $\theta^{\star} = \theta_1$ and $\theta^{\star} = \theta_2$, respectively: while the linear and log-linear rules fail to recover from the attack, Algorithm 4 enables agent 7 to infer the truth. Similar conclusions hold for the other regular agents.

5.7 Chapter Summary

In this chapter, we proposed and analyzed a novel algorithm for addressing the problem of distributed hypothesis testing. The key distinguishing feature of our learning algorithm is that it does not employ any linear consensus-based data aggregation protocol. Instead, it relies on a "min-rule" to spread beliefs through the network. Under mild assumptions of global identifiability and joint strong-connectivity, we established consistency of our learning rule. In particular, we showed that the rate of learning resulting from our approach strictly improves upon all existing rates. For static networks, we established consistency of our algorithm under minimal requirements on the observation model and the network structure. Finally, we proposed a simple and computationally-efficient version of our learning rule that accounts for worst-case adversarial behavior on the part of certain agents in the network.

5.8 Omitted Proofs

5.8.1 Proof of Theorem 5.4.1

The proof of Theorem 5.4.1 is based on several intermediate results. We start with the following simple lemma that characterizes the asymptotic behavior of the local belief sequences generated based on (5.3); we provide a proof (adapted to our notation) to keep the paper self-contained, and to introduce certain quantities that will be referenced later in our analysis.

Lemma 5.8.1 *Consider a false hypothesis $\theta \in \Theta \setminus \{\theta^\star\}$, and an agent $i \in \mathcal{S}(\theta^\star, \theta)$. Suppose $\pi_{i,0}(\theta_p) > 0, \forall \theta_p \in \Theta$. Then, the update rule (5.3) ensures that (i) $\pi_{i,t}(\theta) \to 0$ a.s., (ii) $\pi_{i,\infty}(\theta^\star) \triangleq \lim_{t\to\infty} \pi_{i,t}(\theta^\star)$ exists a.s. and satisfies $\pi_{i,\infty}(\theta^\star) \geq \pi_{i,0}(\theta^\star)$, and (iii) the following holds:*

$$\lim_{t\to\infty} \frac{1}{t} \log \frac{\pi_{i,t}(\theta)}{\pi_{i,t}(\theta^\star)} = -K_i(\theta^\star, \theta) \; a.s. \tag{5.19}$$

Proof Consider any agent $i \in \mathcal{S}(\theta^\star, \theta)$, and define:

$$\rho_{i,t}(\theta) \triangleq \log \frac{\pi_{i,t}(\theta)}{\pi_{i,t}(\theta^\star)}, \quad \lambda_{i,t}(\theta) \triangleq \log \frac{l_i(s_{i,t}|\theta)}{l_i(s_{i,t}|\theta^\star)}. \tag{5.20}$$

Then, based on (5.3), we obtain the following recursion:

$$\rho_{i,t+1}(\theta) = \rho_{i,t}(\theta) + \lambda_{i,t+1}(\theta), \forall t \in \mathbb{N}. \tag{5.21}$$

Rolling out the above equation over time yields

$$\rho_{i,t}(\theta) = \rho_{i,0}(\theta) + \sum_{k=1}^{t} \lambda_{i,k}(\theta), \forall t \in \mathbb{N}_+. \tag{5.22}$$

Notice that $\{\lambda_{i,t}(\theta)\}$ is a sequence of i.i.d. random variables with finite means (see equation (5.1)). In particular, it is easy to verify that each random variable $\lambda_{i,t}(\theta)$ has mean[9] given by $-K_i(\theta^\star, \theta)$. Thus, based on the strong law of large numbers, we have $\frac{1}{t} \sum_{k=1}^{t} \lambda_{i,k}(\theta) \to -K_i(\theta^\star, \theta)$ almost surely. Dividing both sides of (5.22) by t, and taking the limit as t goes to infinity, we then obtain

$$\lim_{t\to\infty} \frac{1}{t} \rho_{i,t}(\theta) = -K_i(\theta^\star, \theta) \; a.s., \tag{5.23}$$

establishing part (iii) of the lemma. Now note that based on the definition of the set $\mathcal{S}(\theta^\star, \theta)$, $K_i(\theta^\star, \theta) > 0$. It then follows from (5.23) that $\rho_{i,t}(\theta) \to -\infty$ almost surely, and hence $\pi_{i,t}(\theta) \to 0$ almost surely. This establishes part (i) of the lemma. For any $\theta \in \Theta_i^{\theta^\star}$, observe that $\lambda_{i,t}(\theta) = 0, \forall t \in \mathbb{N}_+$. It then follows from (5.21) that for each

[9] More precisely, the mean here is obtained by using the expectation operator $\mathbb{E}^{\theta^\star}[\cdot]$ associated with the measure $\mathbb{P}^{\theta^\star}$.

$\theta \in \Theta_i^{\theta^\star}$, $\rho_{i,t}(\theta) = \rho_{i,0}(\theta), \forall t \in \mathbb{N}_+$. From the above discussion, we conclude that a limiting belief vector $\boldsymbol{\pi}_{i,\infty}$ exists almost surely, with non-zero entries corresponding to each $\theta \in \Theta_i^{\theta^\star}$. Part (ii) of the lemma then follows readily. ■

While our proposed learning rule is tailored to facilitate propagation of low beliefs on false hypotheses, it is crucial to also ensure that the beliefs of all agents on the true state remain bounded away from zero. In particular, consider the following scenario. During a transient phase, certain agents see private signals that cause them to temporarily lower their local beliefs on the true state. This effect manifests itself in the actual beliefs of the agents via the min-rule (5.4). We ask: can such a transient phenomenon trigger a cascade of progressively lower beliefs on the true state? The next important result asserts that this will almost surely never be the case.

Lemma 5.8.2 *Suppose the conditions stated in Theorem 5.4.1 hold, and Algorithm 3 is employed by each agent. Then, there exists a set $\bar{\Omega} \subseteq \Omega$ with the following properties: (i) $\mathbb{P}^{\theta^\star}(\bar{\Omega}) = 1$, and (ii) for each $\omega \in \bar{\Omega}$, there exist constants $\eta(\omega) \in (0,1)$ and $t'(\omega) \in (0,\infty)$ such that on the sample path ω,*

$$\pi_{i,t}(\theta^\star) \geq \eta(\omega), \mu_{i,t}(\theta^\star) \geq \eta(\omega), \forall t \geq t'(\omega), \forall i \in \mathcal{V}. \tag{5.24}$$

Proof Let $\bar{\Omega} \subseteq \Omega$ denote the set of sample paths for which assertions (i)-(iii) in Lemma 5.8.1 hold for each false hypothesis $\theta \in \Theta \setminus \{\theta^\star\}$. Based on Lemma 5.8.1, we note that $\mathbb{P}^{\theta^\star}(\bar{\Omega}) = 1$. Consequently, to prove the result, it suffices to establish the existence of $\eta(\omega) \in (0,1)$, and $t'(\omega) \in (0,\infty)$ for each sample path $\omega \in \bar{\Omega}$, such that (5.24) holds. To this end, fix an arbitrary sample path $\omega \in \bar{\Omega}$. We first argue that the local beliefs of every agent on the true state $\theta^\star$ are bounded away from 0 on ω. To see this, pick any agent $i \in \mathcal{V}$. Suppose there exists some $\theta \in \Theta \setminus \{\theta^\star\}$ for which $i \in \mathcal{S}(\theta^\star, \theta)$. Then, based on our choice of ω, Lemma 5.8.1 implies that $\pi_{i,\infty}(\theta^\star) \geq \pi_{i,0}(\theta^\star) > 0$, where the last inequality follows from the requirement of non-zero priors in line 1 of Algo. 3. In particular, given the structure of the update rule (5.3), it follows that $\pi_{i,t}(\theta^\star) > 0$ for all time. This is true since if $\pi_{i,t}(\theta^\star) = 0$ at any instant, then the corresponding belief would remain at 0 for all subsequent time-steps, thereby violating the fact that $\pi_{i,\infty}(\theta^\star) \geq \pi_{i,0}(\theta^\star) > 0$. Now consider the scenario where there exists no $\theta \in \Theta \setminus \{\theta^\star\}$ for which $i \in \mathcal{S}(\theta^\star, \theta)$, i.e., every hypothesis in Θ is observationally equivalent to $\theta^\star$ from the point of view of agent i. In this case, it is easy to see that based on (5.3), $\boldsymbol{\pi}_{i,t} = \boldsymbol{\pi}_{i,0}, \forall t \in \mathbb{N}_+$. In particular, this implies $\pi_{i,t}(\theta^\star) = \pi_{i,0}(\theta^\star) > 0, \forall t \in \mathbb{N}_+$. This establishes our claim that on ω, $\pi_{i,t}(\theta^\star)$ remains bounded away from zero $\forall i \in \mathcal{V}$.

To proceed, define $\gamma_1 \triangleq \min_{i\in\mathcal{V}} \pi_{i,0}(\theta^\star) > 0$, where the inequality follows from line 1 in Algo 3. Pick a small number $\delta > 0$ such that $\delta < \gamma_1$, and notice that our discussion concerning the evolution of the local beliefs readily implies the existence of a time-step $t'(\omega)$, such that for all $t \geq t'(\omega)$, $\pi_{i,t}(\theta^\star) \geq \gamma_1 - \delta > 0, \forall i \in \mathcal{V}$. With $\gamma_2(\omega) \triangleq \min_{i\in\mathcal{V}}\{\mu_{i,t'(\omega)}(\theta^\star)\}$, we claim that $\gamma_2(\omega) > 0$. The claim follows by noting that given the structure of the update rule (5.4), and the requirement of non-zero priors in Algo 3, $\gamma_2(\omega)$ can equal 0 if and only if some agent in the network sets its local belief on $\theta^\star$ to 0 at some time-step prior to $t'(\omega)$. However, this possibility is ruled out in view of the previously established fact that on ω, $\pi_{i,t}(\theta^\star) > 0, \forall t \in \mathbb{N}, \forall i \in \mathcal{V}$. Let $\eta(\omega) = \min\{\gamma_1 - \delta, \gamma_2(\omega)\} > 0$. In words, $\eta(\omega)$ lower-bounds the lowest belief (considering both local and actual beliefs) on the true state $\theta^\star$ held by an agent at time-step $t'(\omega)$. It is apparent from the preceding discussion that $\pi_{i,t}(\theta^\star) \geq \eta(\omega), \forall t \geq t'(\omega), \forall i \in \mathcal{V}$. Thus, to complete the proof, it remains to establish that $\mu_{i,t}(\theta^\star) \geq \eta(\omega), \forall t \geq t'(\omega), \forall i \in \mathcal{V}$. To this end, let us fix an agent i and observe the following:

$$\begin{aligned}
\mu_{i,t'(\omega)+1}(\theta^\star) &\overset{(a)}{=} \frac{\min\{\{\mu_{j,t'(\omega)}(\theta^\star)\}_{j\in\mathcal{N}_i[t'(\omega)]\cup\{i\}}, \pi_{i,t'(\omega)+1}(\theta^\star)\}}{\sum\limits_{p=1}^{m}\min\{\{\mu_{j,t'(\omega)}(\theta_p)\}_{j\in\mathcal{N}_i[t'(\omega)]\cup\{i\}}, \pi_{i,t'(\omega)+1}(\theta_p)\}} \\
&\overset{(b)}{\geq} \frac{\eta(\omega)}{\sum\limits_{p=1}^{m}\min\{\{\mu_{j,t'(\omega)}(\theta_p)\}_{j\in\mathcal{N}_i[t'(\omega)]\cup\{i\}}, \pi_{i,t'(\omega)+1}(\theta_p)\}} \\
&\geq \frac{\eta(\omega)}{\sum\limits_{p=1}^{m}\pi_{i,t'(\omega)+1}(\theta_p)} \overset{(c)}{=} \eta(\omega),
\end{aligned} \tag{5.25}$$

where (a) is given by (5.4), (b) follows from the way $\eta(\omega)$ is defined and by noting that $\pi_{i,t}(\theta^\star) \geq \eta(\omega), \forall t \geq t'(\omega), \forall i \in \mathcal{V}$, and (c) follows by noting that the local belief vectors generated via (5.3) are valid probability distributions over the hypothesis set Θ at each time-step, and hence $\sum\limits_{p=1}^{m}\pi_{i,t'(\omega)+1}(\theta_p) = 1$. The above reasoning applies to every agent in the network, and can be repeated to establish (5.24) via induction. ■

The next result establishes that the intrinsic discriminatory capabilities of an agent are preserved under our learning rule.

Lemma 5.8.3 *Suppose the conditions stated in Theorem 5.4.1 hold, and Algorithm 3 is employed by each agent. Consider any false hypothesis $\theta \in \Theta \setminus \{\theta^\star\}$, and an agent $i \in \mathcal{S}(\theta^\star, \theta)$. Then,*

$$\liminf_{t\to\infty} -\frac{\log \mu_{i,t}(\theta)}{t} \geq K_i(\theta^\star, \theta) \ a.s. \tag{5.26}$$

Proof With $\bar{\Omega}$ defined as in Lemma 5.8.2, recall that $\mathbb{P}^{\theta^\star}(\bar{\Omega}) = 1$, and pick any $\omega \in \bar{\Omega}$. Now consider any false hypothesis $\theta \in \Theta \setminus \{\theta^\star\}$, and an agent $i \in \mathcal{S}(\theta^\star, \theta)$. Fix any $\epsilon > 0$, and notice that since $i \in \mathcal{S}(\theta^\star, \theta)$, Eq. (5.19) in Lemma 5.8.1 implies that there exists $t_i(\omega, \theta, \epsilon)$, such that

$$\pi_{i,t}(\theta) < e^{-(K_i(\theta^\star,\theta)-\epsilon)t}, \forall t \geq t_i(\omega, \theta, \epsilon). \tag{5.27}$$

Furthermore, since $\omega \in \bar{\Omega}$, Lemma 5.8.2 guarantees the existence of a time-step $t'(\omega) \in (0, \infty)$, and a constant $\eta(\omega) \in (0,1)$, such that on ω, $\pi_{i,t}(\theta^\star) \geq \eta(\omega), \mu_{i,t}(\theta^\star) \geq \eta(\omega), \forall t \geq t'(\omega), \forall i \in \mathcal{V}$. Let $\bar{t}_i(\omega, \theta, \epsilon) = \max\{t'(\omega), t_i(\omega, \theta, \epsilon)\}$. Let us suppress the dependence of $\bar{t}_i(\omega, \theta, \epsilon)$ on i, ω, θ and ϵ for simplicity of notation, and observe the following inequalities:

$$\begin{aligned}
\mu_{i,\bar{t}+1}(\theta) &\overset{(a)}{\leq} \frac{\pi_{i,\bar{t}+1}(\theta)}{\sum\limits_{p=1}^{m} \min\{\{\mu_{j,\bar{t}}(\theta_p)\}_{j \in \mathcal{N}_i[\bar{t}] \cup \{i\}}, \pi_{i,\bar{t}+1}(\theta_p)\}} \\
&\leq \frac{\pi_{i,\bar{t}+1}(\theta)}{\min\{\{\mu_{j,\bar{t}}(\theta^\star)\}_{j \in \mathcal{N}_i[\bar{t}] \cup \{i\}}, \pi_{i,\bar{t}+1}(\theta^\star)\}} \\
&\overset{(b)}{<} \frac{e^{-(K_i(\theta^\star,\theta)-\epsilon)(\bar{t}+1)}}{\eta(\omega)}.
\end{aligned} \tag{5.28}$$

In the above inequalities, (a) follows from (5.4), whereas (b) follows from (5.27) and by noting that all agents have both their local and actual beliefs lower bounded by $\eta(\omega)$ beyond time-step $\bar{t}$. In particular, it is easy to see that the arguments used to arrive at (5.28) apply to each time-step $t \geq \bar{t} + 1$. Based on (5.28), we then obtain that $\forall t \geq \bar{t} + 1$:

$$-\frac{\log \mu_{i,t}(\theta)}{t} > (K_i(\theta^\star, \theta) - \epsilon) + \frac{\log \eta(\omega)}{t}. \tag{5.29}$$

Taking the limit inferior on both sides of (5.29), and noting that ϵ can be made arbitrarily small, readily leads to (5.26). ■

For the subsequent discussion, let us fix a particular false hypothesis $\theta \in \Theta \setminus \{\theta^\star\}$, and assume that global identifiability holds. Let $v_\theta \in \operatorname{argmax}_{l \in \mathcal{S}(\theta^\star, \theta)} K_l(\theta^\star, \theta)$ represent any agent with the best discriminatory power w.r.t. the false hypothesis θ, given that $\theta^\star$ gets realized. Based on Lemma 5.8.3, we have

$$\liminf_{t \to \infty} -\frac{\log \mu_{v_\theta,t}(\theta)}{t} \geq K_{v_\theta}(\theta^\star, \theta) \; a.s. \tag{5.30}$$

Our goal is to now establish that each agent $i \in \mathcal{V} \setminus \{v_\theta\}$ inherits the same asymptotic rate of rejection of θ as that of agent v_θ in (5.30). Roughly speaking, we will achieve this by showing that under the assumption of joint strong-connectivity, the belief of any agent $i \in \mathcal{V} \setminus \{v_\theta\}$ on θ is "not too far off" from the belief of agent v_θ on θ. In what follows, we make this idea precise. First, we require some additional notation: with each agent $i \in \mathcal{V}$, we associate a non-negative scalar $c_{i,t}(\theta) \in [0, \infty]$. These parameters evolve based on the following rules.[10]

(i) $c_{v_\theta,t}(\theta) = 0, \forall t \in \mathbb{N}$.

(ii) $c_{i,0}(\theta) = \infty, \forall i \in \mathcal{V} \setminus \{v_\theta\}$.

(iii) For each $i \in \mathcal{V} \setminus \{v_\theta\}$ and $t \in \mathbb{N}$, define $\tau_{i,t}(\theta) \triangleq \min_{j \in \mathcal{N}_i[t] \cup \{i\}} c_{j,t}(\theta)$, and

$$c_{i,t+1}(\theta) \triangleq \tau_{i,t}(\theta) + 1. \tag{5.31}$$

To explain the purpose of the above rules, we will adhere to the following terminology. We say that there exists a path of length $m \in \mathbb{N}_+$ from v_θ to $i \in \mathcal{V} \setminus \{v_\theta\}$ over $[t-m, t-1]$, if there exist agents $x(t-m+1), \ldots, x(t) \in \mathcal{V} \setminus \{v_\theta\}$, such that $(x(\tau-1), x(\tau)) \in \mathcal{E}[\tau-1]$, where $\tau \in \{t-m+1, \ldots, t\}, x(t-m) = v_\theta$, and $x(t) = i$. Note that the agents appearing in the path need not be distinct, and that we have assumed the presence of self-loops in each graph $\mathcal{G}[t], t \in \mathbb{N}$. Rules (i)-(iii) have been designed in a manner such that if $c_{i,t}(\theta)$ is finite at any time-step $t \in \mathbb{N}$ for any agent $i \in \mathcal{V} \setminus \{v_\theta\}$, then there exists a path of length $c_{i,t}(\theta)$ from v_θ to i over $[t - c_{i,t}(\theta), t-1]$, in the sense described above. Analyzing the time-evolution of $c_{i,t}(\theta)$ enables us to then relate the belief $\mu_{i,t}(\theta)$ of agent i to a delayed-version of the belief $\mu_{v_\theta,t}(\theta)$ of agent v_θ, where the delay is precisely $c_{i,t}(\theta)$ (the above statements are formalized and proven in Lemma 5.8.5). Since agent v_θ is the reference agent here, its delay w.r.t. its own belief on θ is set to 0 for all time, thus explaining rule (i). Initially, all agents in $\mathcal{V} \setminus \{v_\theta\}$ start out with an "infinite-delay " w.r.t. the belief of agent v_θ; this is captured by rule (ii). Finally, the rationale behind updating $c_{i,t}(\theta)$ via rule (iii) is to formalize the intuition that under the assumption of joint strong-connectivity, the lengths of paths linking v_θ to agents in $\mathcal{V} \setminus \{v_\theta\}$ (and hence, the corresponding delays) should eventually remain uniformly bounded; we begin by establishing this fact in the following lemma.

[10] Note that the agents do not actually maintain or update the parameters $c_{i,t}(\theta)$. Instead, they have been introduced solely for the purpose of analysis.

Lemma 5.8.4 *Consider any $\theta \in \Theta \setminus \{\theta^\star\}$ and suppose the joint strong-connectivity assumption (Assumption 5.2.3) holds. Then, the following is true:*

$$c_{i,t}(\theta) \leq 2(n-1)T, \forall i \in \mathcal{V}, \forall t \geq (n-1)T, \tag{5.32}$$

where T is the constant appearing in Assumption 5.2.3.

Proof Observe that the conclusion in (5.32) is trivially true for agent v_θ since $c_{v_\theta,t}(\theta) = 0, \forall t \in \mathbb{N}$. To prove the result for agents in the set $\mathcal{V} \setminus \{v_\theta\}$, we begin by claiming that

$$c_{i,(n-1)T}(\theta) \leq (n-1)T, \forall i \in \mathcal{V}. \tag{5.33}$$

To prove this claim, let $\mathcal{L}_0(\theta^\star, \theta) = \{v_\theta\}$, and define

$$\mathcal{L}_1(\theta^\star, \theta) \triangleq \{i \in \mathcal{V} \setminus \mathcal{L}_0(\theta^\star, \theta) : \{\bigcup_{\tau=0}^{T-1} \mathcal{N}_i[\tau]\} \cap \mathcal{L}_0(\theta^\star, \theta) \neq \emptyset\} \tag{5.34}$$

as the set of agents in $\mathcal{V} \setminus \{v_\theta\}$ that have a direct edge from agent v_θ at least once over the interval $[0, T)$. Assumption 5.2.3 implies that $\mathcal{L}_1(\theta^\star, \theta)$ is non-empty (barring the trivial case when $\mathcal{V} = \{v_\theta\}$). Now pick any agent $i \in \mathcal{L}_1(\theta^\star, \theta)$, and notice that since $v_\theta \in \mathcal{N}_i[\tau]$ for some $\tau \in [0, T)$, update rule (5.31) implies $c_{i,\tau+1}(\theta) = 1$.[11] In particular, based on (5.31),

$$c_{i,t+1}(\theta) \leq c_{i,t}(\theta) + 1. \tag{5.35}$$

Based on the above discussion, it follows that for each agent $i \in \mathcal{L}_1(\theta^\star, \theta), c_{i,T}(\theta) \leq T$. The claim in (5.33) follows readily for each agent $i \in \mathcal{L}_1(\theta^\star, \theta)$ by appealing to (5.35). Let us now recursively define the sets $\mathcal{L}_r(\theta^\star, \theta), 1 \leq r \leq (n-1)$, as

$$\mathcal{L}_r(\theta^\star, \theta) \triangleq \{i \in \mathcal{V} \setminus \bigcup_{q=0}^{(r-1)} \mathcal{L}_q(\theta^\star, \theta) : \{\bigcup_{\tau=(r-1)T}^{rT-1} \mathcal{N}_i[\tau]\} \cap \{\bigcup_{q=0}^{(r-1)} \mathcal{L}_q(\theta^\star, \theta)\} \neq \emptyset\}. \tag{5.36}$$

In words, $\mathcal{L}_r(\theta^\star, \theta)$ are those agents belonging to $\mathcal{V} \setminus \bigcup_{q=0}^{(r-1)} \mathcal{L}_q(\theta^\star, \theta)$ that each have at least one neighbor from the set $\bigcup_{q=0}^{(r-1)} \mathcal{L}_q(\theta^\star, \theta)$ over the interval $[(r-1)T, rT-1]$.

[11] Notice that based on the update rule (5.31), $c_{i,t}(\theta) \geq 1, \forall i \in \mathcal{V} \setminus \{v_\theta\}$. Thus, $\text{argmin}_{j \in \mathcal{N}_i[t] \cup \{i\}} c_{j,t}(\theta) = v_\theta$ whenever $v_\theta \in \mathcal{N}_i[t]$, since $c_{v_\theta,t}(\theta) = 0, \forall t \in \mathbb{N}$.

We complete the proof of the claim by inducting on r. The base case with $r = 1$ has already been proven above. Now suppose the following is true: $c_{i,rT}(\theta) \leq rT, \forall i \in \mathcal{L}_r(\theta^\star, \theta)$, where $r \in \{1, \ldots, m-1\}$, and $m \in \{2, \ldots, n-1\}$. Let $r = m$. If $\mathcal{V} \setminus \bigcup_{q=0}^{(m-1)} \mathcal{L}_q(\theta^\star, \theta)$ is empty, then we are done. Else, based on Assumption 5.2.3, it must be that $\mathcal{L}_m(\theta^\star, \theta)$ is non-empty. Pick any agent $i \in \mathcal{L}_m(\theta^\star, \theta)$, and notice that it has a neighbor j (say) from the set $\bigcup_{q=0}^{(m-1)} \mathcal{L}_q(\theta^\star, \theta)$ at some time-step $\tau \in [(m-1)T, mT)$. The induction hypothesis coupled with (5.35) implies that $c_{j,\tau}(\theta) \leq \tau$, and hence $c_{i,\tau+1}(\theta) \leq c_{j,\tau}(\theta) + 1 \leq \tau + 1$ based on (5.31). Appealing to (5.35) then reveals that $c_{i,mT}(\theta) \leq mT$, thus completing the induction step. Finally, noting that $\bigcup_{q=0}^{(n-1)} \mathcal{L}_q(\theta^\star, \theta) = \mathcal{V}$ completes our proof of the claim (5.33). An identical line of argument as above can be employed to show that $c_{i,2(n-1)T} \leq (n-1)T, \forall i \in \mathcal{V}$. In particular, this can be done by first taking $\mathcal{C}_0(\theta^\star, \theta) = \{v_\theta\}$, and recursively defining the sets $\mathcal{C}_r(\theta^\star, \theta), 1 \leq r \leq (n-1)$ as

$$\mathcal{C}_r(\theta^\star, \theta) \triangleq \{i \in \mathcal{V} \setminus \bigcup_{q=0}^{(r-1)} \mathcal{C}_q(\theta^\star, \theta) : \{ \bigcup_{\tau=(n+r-2)T}^{(n+r-1)T-1} \mathcal{N}_i[\tau]\} \cap \{ \bigcup_{q=0}^{(r-1)} \mathcal{C}_q(\theta^\star, \theta)\} \neq \emptyset\}. \tag{5.37}$$

One can then easily prove via induction that $c_{i,(n-1+r)T}(\theta) \leq rT, \forall i \in \mathcal{C}_r(\theta^\star, \theta)$, where $1 \leq r \leq (n-1)$. The rest then follows from (5.35).

We can keep repeating the above argument to establish that $c_{i,m(n-1)T}(\theta) \leq (n-1)T, \forall i \in \mathcal{V}, \forall m \in \mathbb{N}_+$. Finally, based on the above bound and (5.35), it follows that for each agent $i \in \mathcal{V}$, $c_{i,t}(\theta)$ is upper-bounded by $2(n-1)T$ at any time-step $t \in (m(n-1)T, (m+1)(n-1)T)$, where $m \in \mathbb{N}_+$. This establishes (5.32) and completes the proof. ■

The next lemma relates $\mu_{i,t}(\theta), i \in \mathcal{V} \setminus \{v_\theta\}$ to $\mu_{v_\theta,t}(\theta)$ in terms of the parameter $c_{i,t}(\theta)$ and, in turn, provides the final ingredient required to prove Theorem 5.4.1.

Lemma 5.8.5 *Consider any $\theta \in \Theta \setminus \{\theta^\star\}$. Suppose the joint strong-connectivity assumption holds (Assumption 5.2.3), and each agent applies Algorithm 3. Suppose $c_{i,t}(\theta)$ is finite, where $i \in \mathcal{V} \setminus \{v_\theta\}$, and $t \in \mathbb{N}$. Then, the following are true.*

(i) There exists a path of length $c_{i,t}(\theta)$ from v_θ to i over $[t - c_{i,t}(\theta), t-1]$.

(ii) Let the path linking v_θ to i over $[t - c_{i,t}(\theta), t-1]$ in part (i) be denoted $x(t - c_{i,t}(\theta)), x(t - c_{i,t}(\theta)+1), \ldots, x(t)$, where $x(t - c_{i,t}(\theta)) = v_\theta$ and $x(t) = i$. Then

$$\mu_{i,t}(\theta) \leq \frac{\mu_{v_\theta, a_{i,t}(\theta)}(\theta)}{\prod\limits_{\tau = a_{i,t}(\theta)+1}^{t} \eta_{x(\tau),\tau}(\theta^\star)}, \tag{5.38}$$

where $a_{i,t}(\theta) = t - c_{i,t}(\theta)$, and

$$\eta_{i,t}(\theta^\star) \triangleq \min\{\{\mu_{j,t-1}(\theta^\star)\}_{j \in \mathcal{N}_i[t-1]\cup\{i\}}, \pi_{i,t}(\theta^\star)\}, \forall i \in \mathcal{V}. \tag{5.39}$$

Proof We prove part (i) by inducting on the value of $c_{i,t}(\theta)$. For the base case, suppose $c_{i,t}(\theta) = 1$ for some agent $i \in \mathcal{V} \setminus \{v_\theta\}$ at some time-step t. Based on (5.31), notice that this can happen if and only if $v_\theta \in \mathcal{N}_i[t-1]$; the claim in part (i) then follows readily for the base case. Fix an integer $m \geq 2$, and suppose that the assertion of part (i) holds for any agent $i \in \mathcal{V} \setminus \{v_\theta\}$ and at any time-step t, whenever $c_{i,t}(\theta) \in \{1, \ldots, m-1\}$. Now suppose that at some time-step t, $c_{i,t}(\theta) = m$ for some agent $i \in \mathcal{V} \setminus \{v_\theta\}$. Referring to (5.31), this is true only if $c_{l,t-1}(\theta) = m-1$ for some $l \in \mathcal{N}_i[t-1] \cup \{i\}$. Since $m \geq 2$, we have $c_{l,t-1}(\theta) \geq 1$, and hence $l \in \mathcal{V} \setminus \{v_\theta\}$. The induction hypothesis thus applies to agent l, implying the existence of a path of length $m-1$ from v_θ to l over $[(t-1) - c_{l,t-1}(\theta), t-2]$, i.e., over $[t-m, t-2]$. Appending this path with the edge $(l, i) \in \mathcal{E}[t-1]$ immediately leads to the desired conclusion.

For part (ii), consider the path $x(t - c_{i,t}(\theta)), x(t - c_{i,t}(\theta)+1), \ldots, x(t)$ from v_θ to i over $[t - c_{i,t}(\theta), t-1]$, where $x(t - c_{i,t}(\theta)) = v_\theta$ and $x(t) = i$. By definition of this path, $x(\tau - 1) \in \mathcal{N}_{x(\tau)}[\tau - 1] \cup \{x(\tau)\}$, for all $\tau \in \{a_{i,t}(\theta)+1, \ldots, t\}$. Thus, referring to (5.4), we obtain

$$\begin{aligned} \mu_{x(\tau),\tau}(\theta) &\leq \frac{\mu_{x(\tau-1),\tau-1}(\theta)}{\sum\limits_{p=1}^{m} \min\{\{\mu_{j,\tau-1}(\theta_p)\}_{j \in \mathcal{N}_{x(\tau)}[\tau-1]\cup\{x(\tau)\}}, \pi_{x(\tau),\tau}(\theta_p)\}} \\ &\leq \frac{\mu_{x(\tau-1),\tau-1}(\theta)}{\eta_{x(\tau),\tau}(\theta^\star)}. \end{aligned} \tag{5.40}$$

Using the above inequality recursively with $\tau \in \{a_{i,t}(\theta)+1,\ldots,t\}$ immediately leads to (5.38). ■

Proof (Theorem 5.4.1): Fix a false hypothesis $\theta \in \Theta \setminus \{\theta^\star\}$. Based on the assumption of global identifiability, note that the set $\mathcal{S}(\theta^\star,\theta)$ is non-empty. Recall that v_θ is any agent for which $K_i(\theta^\star,\theta), i \in \mathcal{S}(\theta^\star,\theta)$ is maximum, and note that we have already established that the assertion of Theorem 5.4.1, namely inequality (5.5), holds for agent v_θ in Lemma 5.8.3. Now consider an agent $i \in \mathcal{V}\setminus\{v_\theta\}$, and notice that if $t \geq (n-1)T$, then $c_{i,t}(\theta)$ is uniformly bounded based on Lemma 5.8.4. Thus, the assertions in Lemma 5.8.5 hold for all $t \geq (n-1)T$. Taking the natural log on both sides of (5.38), dividing throughout by t, and simplifying, we obtain the following for all $t \geq (n-1)T$:

$$-\frac{\log \mu_{i,t}(\theta)}{t} \geq -\frac{\log \mu_{v_\theta,a_{i,t}(\theta)}(\theta)}{t} + \sum_{\tau=a_{i,t}(\theta)+1}^{t} \frac{\log \eta_{x(\tau),\tau}(\theta^\star)}{t}, \tag{5.41}$$

where $a_{i,t}(\theta) = t - c_{i,t}(\theta)$, $\eta_{i,t}(\theta^\star)$ is as defined in (5.39), and $x(\tau), \tau \in \{a_{i,t}(\theta)+1,\ldots,t\}$, are agents in the path linking v_θ to i over $[a_{i,t}(\theta), t-1]$. For the remainder of the proof, to lighten the notation, let us drop the subscript on v_θ, and let $a(t) = a_{i,t}(\theta)$. Based on (5.4), we then have:

$$\mu_{v,a(t)}(\theta) \leq \frac{\pi_{v,a(t)}(\theta)}{\eta_{v,a(t)}(\theta^\star)}. \tag{5.42}$$

A bit of straightforward algebra then yields:

$$-\frac{\log \mu_{v,a(t)}(\theta)}{t} \geq -\frac{\log \pi_{v,t}(\theta)}{t} + \frac{\log \frac{\pi_{v,t}(\theta)}{\pi_{v,a(t)}(\theta)}}{t} + \frac{\log \eta_{v,a(t)}(\theta^\star)}{t}. \tag{5.43}$$

Combining (5.41) and (5.43), we obtain for $t \geq (n-1)T$:

$$-\frac{\log \mu_{i,t}(\theta)}{t} \geq -\frac{\log \pi_{v,t}(\theta)}{t} + b(t), \tag{5.44}$$

where $b(t) = b_1(t) + b_2(t) + b_3(t)$,

$$b_1(t) = \sum_{\tau=a(t)+1}^{t} \frac{\log \eta_{x(\tau),\tau}(\theta^\star)}{t},\; b_2(t) = \frac{\log \frac{\pi_{v,t}(\theta)}{\pi_{v,a(t)}(\theta)}}{t}, \tag{5.45}$$

and

$$b_3(t) = \frac{\log \eta_{v,a(t)}(\theta^\star)}{t}. \tag{5.46}$$

We now argue that each of the terms $b_1(t), b_2(t)$ and $b_3(t)$ converge to 0 almost surely as $t \to \infty$. To do so, recall that the set $\bar{\Omega} \subseteq \Omega$ in Lemma 5.8.2 has measure 1. In what follows, we prove that $b_1(t), b_2(t)$ and $b_3(t)$ converge to 0 for each sample path $\omega \in \bar{\Omega}$. Accordingly, fix $\omega \in \bar{\Omega}$, and recall $\eta(\omega) \in (0,1)$ and $t'(\omega) \in (0,\infty)$ from Lemma 5.8.2. Suppose $t > t'(\omega) + 2\bar{T}$, where $\bar{T} = (n-1)T$. We then claim the following:

$$\pi_{l,\tau}(\theta^\star) \geq \eta(\omega), \mu_{l,\tau}(\theta^\star) \geq \eta(\omega), \forall l \in \mathcal{V}, \forall \tau \geq a(t). \tag{5.47}$$

To see why this is true, notice that based on Lemma 5.8.4, the following holds when $t > t'(\omega) + 2\bar{T}$:

$$a(t) = t - c_{i,t}(\theta) \geq t - 2\bar{T} > t'(\omega). \tag{5.48}$$

The claim regarding (5.47) then follows readily from equation (5.24) in Lemma 5.8.2. Based on the above discussion, and referring to (5.39), we immediately note that when $t > t'(\omega) + 2\bar{T}$,

$$\eta_{l,\tau}(\theta^\star) \geq \eta(\omega), \forall l \in \mathcal{V}, \forall \tau \geq a(t). \tag{5.49}$$

For establishing the convergence of $b_1(t), b_2(t)$ and $b_3(t)$, suppose $t > t'(\omega) + 2\bar{T}$. Regarding $b_1(t)$, we then observe:

$$\begin{aligned} |b_1(t)| &= \left| \sum_{\tau=a(t)+1}^{t} \frac{\log \eta_{x(\tau),\tau}(\theta^\star)}{t} \right| \\ &\overset{(a)}{\leq} \sum_{\tau=a(t)+1}^{t} \frac{\left|\log \eta_{x(\tau),\tau}(\theta^\star)\right|}{t} \\ &\overset{(b)}{\leq} \frac{(t-a(t))}{t} \log \frac{1}{\eta(\omega)} \\ &\overset{(c)}{\leq} \frac{2\bar{T}}{t} \log \frac{1}{\eta(\omega)}, \end{aligned} \tag{5.50}$$

where (a) follows from the triangle inequality, (b) follows from (5.49), and (c) follows from (5.48). From (5.50), we immediately note that $b_1(t) \to 0$ along ω. Let us now turn our attention to $b_2(t)$, and take note of the following:

$$\begin{aligned} |b_2(t)| &\overset{(a)}{=} \frac{1}{t}\left|\log \frac{\pi_{v,t}(\theta^\star)}{\pi_{v,a(t)}(\theta^\star)} + \sum_{\tau=a(t)+1}^{t} \log \frac{l_v(s_{v,\tau}|\theta)}{l_v(s_{v,\tau}|\theta^\star)}\right| \\ &\overset{(b)}{\leq} \frac{1}{t}\left|\log \frac{\pi_{v,t}(\theta^\star)}{\pi_{v,a(t)}(\theta^\star)}\right| + \frac{1}{t}\sum_{\tau=a(t)+1}^{t} \left|\log \frac{l_v(s_{v,\tau}|\theta)}{l_v(s_{v,\tau}|\theta^\star)}\right| \\ &\overset{(c)}{\leq} \frac{2}{t}\log\frac{1}{\eta(\omega)} + \frac{(t-a(t))L}{t} \\ &\overset{(d)}{\leq} \frac{2}{t}\left(\log\frac{1}{\eta(\omega)} + L\bar{T}\right), \end{aligned} \tag{5.51}$$

where (a) follows from (5.22) and some simple manipulations, (b) is a consequence of the triangle inequality, (c) follows from (5.1) and (5.47), and (d) follows from (5.48). Based on (5.51), we then note that $b_2(t) \to 0$ along ω. Finally, the fact that $b_3(t)$ converges to 0 along ω follows immediately by appealing to (5.49). We have thus established that $b(t) \to 0$ almost surely. The desired conclusion then follows by taking the limit inferior on both sides of (5.44), and noting that

$$\lim_{t\to\infty} -\frac{\log \pi_{v,t}(\theta)}{t} = \lim_{t\to\infty} -\frac{1}{t}\rho_{v,t}(\theta) = K_v(\theta^\star, \theta) \; a.s., \tag{5.52}$$

where $\rho_{v,t}(\theta)$ is as defined in Lemma 5.8.1. The fact that $\mu_{i,t}(\theta) \to 0$ is immediate, since $K_v(\theta^\star, \theta) > 0$ based on global identifiability. The above analysis applies identically to each $\theta \in \Theta \setminus \{\theta^\star\}$. This establishes consistency of our rule, and completes the proof. ■

5.8.2 Proof of Theorem 5.4.2

To prove Theorem 5.4.2, we will make use of one of Littlewood's three principles: every pointwise convergent sequence of measurable functions is nearly uniformly convergent.

Theorem 5.8.6 *(**Egoroff's Theorem**) [154, Chapter 18] Let $(X, \mathcal{M}, \mu)$ be a finite measure space and $\{f_n\}$ a sequence of measurable functions on X that converge pointwise a.e. (almost everywhere) on X to a function f that is finite a.e. on X. Then* *for each $\epsilon > 0$, there is a measurable subset X_ϵ of X for which $f_n \to f$ uniformly on X_ϵ, and $\mu(X_\epsilon) \geq 1 - \epsilon$.*

Proof (**Theorem 5.4.2**): Consider a $\theta \in \Theta \setminus \{\theta^\star\}$, and recall that $K_{v_\theta}(\theta^\star, \theta) = \max_{l \in \mathcal{S}(\theta^\star,\theta)} K_l(\theta^\star, \theta) = \bar{K}(\theta^\star, \theta)$. We only prove the result for $i \in \mathcal{V} \setminus \{v_\theta\}$, since the argument for agent v_θ will be similar. To this end, let us fix an agent $i \in \mathcal{V} \setminus \{v_\theta\}$. We adhere to the notation used in the proof of Lemma 5.8.1, and for simplicity assume that the initial local belief vectors $\boldsymbol{\pi}_{i,0}, i \in \mathcal{V}$ are uniform distributions over the hypothesis set Θ; our subsequent arguments will continue to hold (with simple modifications) under the more general assumption on priors in line 1 of Algo 3. We immediately note that based on the assumption of uniform priors, $\rho_{i,0}(\theta) = 0, \forall i \in \mathcal{V}$. Now referring to inequality (5.44) in the proof of Theorem 5.4.1, we obtain the following for $t \geq (n-1)T$:

$$\begin{aligned} &\mathbb{P}^{\theta^\star}\left(-\frac{\log \mu_{i,t}(\theta)}{t} \leq \bar{K}(\theta^\star, \theta) - \frac{\epsilon}{2} + b(t)\right) \\ &\overset{(a)}{\leq} \mathbb{P}^{\theta^\star}\left(-\frac{\log \pi_{v_\theta,t}(\theta)}{t} \leq \bar{K}(\theta^\star, \theta) - \frac{\epsilon}{2}\right) \\ &\overset{(b)}{\leq} \mathbb{P}^{\theta^\star}\left(-\frac{\rho_{v_\theta,t}(\theta)}{t} \leq \bar{K}(\theta^\star, \theta) - \frac{\epsilon}{2}\right) \\ &\overset{(c)}{=} \mathbb{P}^{\theta^\star}\left(\frac{1}{t}\sum_{k=1}^{t} \lambda_{v_\theta,k}(\theta) - (-K_{v_\theta}(\theta^\star, \theta)) \geq \frac{\epsilon}{2}\right) \\ &\overset{(d)}{\leq} \exp(-\frac{\epsilon^2 t}{8L^2}). \end{aligned} \tag{5.53}$$

In the above steps, (a) follows directly from (5.44), and (b) follows by noting that based on the definition of $\rho_{v_\theta,t}(\theta)$,

$$\frac{\log \pi_{v_\theta,t}(\theta)}{t} \leq \frac{\rho_{v_\theta,t}(\theta)}{t}, \forall t \in \mathbb{N}. \tag{5.54}$$

Step (c) follows directly from (5.22) with $\rho_{v_\theta,0}(\theta) = 0$. Finally, noting that $\frac{1}{t}\sum\limits_{k=1}^{t} \lambda_{v_\theta,k}(\theta) \to -K_{v_\theta}(\theta^\star, \theta)$ a.s. (as argued in the proof of Lemma 5.8.1), using the fact that $|\lambda_{v_\theta,t}(\theta)| \leq L, \forall t \in \mathbb{N}_+$ based on (5.1), and applying Hoeffding's inequality [155, Theorem 2], leads to (d). Now recall from the proof of Theorem 5.4.1 that $b(t) \to 0$ almost surely. Appealing to Egoroff's theorem, we then infer that given any arbitrarily small $\delta \in (0, 1)$, there exists a set $\Omega'(\delta) \subseteq \Omega$ of $\mathbb{P}^{\theta^\star}$-measure at least $(1-\delta)$, such that $b(t)$ converges to 0 uniformly on $\Omega'(\delta)$. Thus, given any $\epsilon > 0$, there exists a ω-independent constant $t(\epsilon, \delta) \in (0, \infty)$, such that $|b(t)| \leq \frac{\epsilon}{2}, \forall t \geq t(\epsilon, \delta)$, along each sample path $\omega \in \Omega'(\delta)$. Setting $t'(\epsilon, \delta, n, T) = \max\{t(\epsilon, \delta), (n-1)T\}$, and referring to (5.53), we immediately obtain that $\forall t \geq t'(\epsilon, \delta, n, T)$,

$$\begin{aligned} &\mathbb{P}^{\theta^\star}\left(\left\{-\frac{\log \mu_{i,t}(\theta)}{t} \leq \bar{K}(\theta^\star, \theta) - \epsilon\right\} \cap \Omega'(\delta)\right) \\ &\leq \mathbb{P}^{\theta^\star}\left(\left\{-\frac{\log \mu_{i,t}(\theta)}{t} \leq \bar{K}(\theta^\star, \theta) - \frac{\epsilon}{2} + b(t)\right\} \cap \Omega'(\delta)\right) \\ &\leq \mathbb{P}^{\theta^\star}\left(-\frac{\log \mu_{i,t}(\theta)}{t} \leq \bar{K}(\theta^\star, \theta) - \frac{\epsilon}{2} + b(t)\right) \leq \exp(-\frac{\epsilon^2 t}{8L^2}). \end{aligned} \tag{5.55}$$

Taking the natural log on both sides of the resulting inequality, dividing throughout by t, simplifying, and then taking the limit inferior on both sides, leads to the desired result. ■

5.8.3 Proof of Theorem 5.5.1

Proof Consider an f-local adversarial set $\mathcal{A} \subset \mathcal{V}$, and let $\mathcal{R} = \mathcal{V} \setminus \mathcal{A}$. We study two separate cases.

Case 1: Consider a regular agent $i \in \mathcal{R}$ such that $|\mathcal{N}_i| < (2f+1)$. Based on the hypothesis of the theorem, we claim that $i \in \mathcal{S}(\theta_p, \theta_q)$, for every pair $\theta_p, \theta_q \in \Theta$. We prove this claim via contradiction. To do so, suppose there exists a pair $\theta_p, \theta_q \in \Theta$, such that $i \in \mathcal{V} \setminus \mathcal{S}(\theta_p, \theta_q)$. As $|\mathcal{N}_i| < (2f+1)$, the set $\{i\}$ is clearly not $(2f+1)$-reachable (see Def. 3.7.1). Thus, $\mathcal{G}$ is not strongly $(2f+1)$-robust w.r.t. the source set $\mathcal{S}(\theta_p, \theta_q)$, a fact that contradicts the hypothesis of the theorem. Thus, we have established that if the graph-theoretic condition identified in the theorem is met, then regular agents with fewer than $(2f+1)$ neighbors can distinguish between every pair of hypotheses. For such agents, the assertion of the theorem then follows directly from Lemma 5.8.1, and update rules (5.3) and (5.14).

Case 2: We now focus only on regular agents i satisfying $|\mathcal{N}_i| \geq (2f+1)$. A key property of the LFRHE algorithm (Algo. 4) that will be used throughout the proof is as follows. For any $i \in \mathcal{R}$, and any $\theta \in \Theta$, the filtering operation in line 7 of Algo. 4 ensures that at each $t \in \mathbb{N}$, we have

$$\mu_{j,t}(\theta) \in Conv(\Psi^{\theta}_{i,t}), \forall j \in \mathcal{M}^{\theta}_{i,t}, \tag{5.56}$$

where

$$\Psi^{\theta}_{i,t} \triangleq \{\mu_{l,t}(\theta) : l \in \mathcal{N}_i \cap \mathcal{R}\}, \tag{5.57}$$

and $Conv(\Psi^{\theta}_{i,t})$ is used to denote the convex hull formed by the points in the set $\Psi^{\theta}_{i,t}$ (recall that $\mathcal{M}^{\theta}_{i,t}$ was defined in line 8 of Algo 4 to be the set of agents in $\mathcal{N}_i$ whose beliefs are retained by agent i after it removes the highest f and lowest f beliefs $\mu_{j,t}(\theta), j \in \mathcal{N}_i$). In words, any neighboring belief (on a particular hypothesis) that agent i uses in the update rule (5.13) lies in the convex hull of the actual beliefs of its regular neighbors (on that particular hypothesis). To see why (5.56) is true, partition the neighbor set $\mathcal{N}_i$ of a regular agent into three sets $\mathcal{U}^{\theta}_{i,t}, \mathcal{M}^{\theta}_{i,t}$, and $\mathcal{J}^{\theta}_{i,t}$ as follows. Sets $\mathcal{U}^{\theta}_{i,t}$ and $\mathcal{J}^{\theta}_{i,t}$ are each of cardinality f, and contain neighbors of agent i that transmit the highest f and the lowest f actual beliefs respectively, on the hypothesis θ, to agent i at time-step t. The set $\mathcal{M}^{\theta}_{i,t}$ contains the remaining neighbors of agent i, and is non-empty at every time-step since $|\mathcal{N}_i| \geq (2f+1)$. If $\mathcal{M}^{\theta}_{i,t} \cap \mathcal{A} = \emptyset$, then (5.56) holds trivially. Thus, consider the case when there are adversaries in the set $\mathcal{M}^{\theta}_{i,t}$, i.e., $\mathcal{M}^{\theta}_{i,t} \cap \mathcal{A} \neq \emptyset$. Given the f-locality of the adversarial model, and the nature of the filtering operation in the LFRHE algorithm, we infer that for each $j \in \mathcal{M}^{\theta}_{i,t} \cap \mathcal{A}$, there exist regular agents $u, v \in \mathcal{N}_i \cap \mathcal{R}$, such that $u \in \mathcal{U}^{\theta}_{i,t}$, $v \in \mathcal{J}^{\theta}_{i,t}$, and $\mu_{v,t}(\theta) \leq \mu_{j,t}(\theta) \leq \mu_{u,t}(\theta)$. This establishes our claim regarding equation (5.56).

With the above property in hand, let $\bar{\Omega} \subseteq \Omega$ denote the set of sample paths for which assertions (i)-(iii) in Lemma 5.8.1 hold when restricted to the set of regular agents $\mathcal{R}$. Since the evolution of the local beliefs are unaffected by the presence of adversaries, Lemma 5.8.1 implies $\mathbb{P}^{\theta^{\star}}(\bar{\Omega}) = 1$. Now as in Lemma 5.8.2, fix a sample path $\omega \in \bar{\Omega}$. Define $\gamma_1 \triangleq \min_{i \in \mathcal{R}} \pi_{i,0}(\theta^{\star})$, pick a small number $\delta > 0$ satisfying $\delta < \gamma_1$, and observe that arguments similar to those in the proof of Lemma 5.8.2 imply the existence of a time-step $t'(\omega)$, such that for all $t \geq t'(\omega), \pi_{i,t}(\theta^{\star}) \geq \gamma_1 - \delta > 0, \forall i \in \mathcal{R}$.

Let $\gamma_2(\omega) \triangleq \min_{i \in \mathcal{R}}\{\mu_{i,t'(\omega)}(\theta^\star)\}$. As before, we claim $\gamma_2(\omega) > 0$. To establish this claim, we need to answer the following question: can an adversarial agent cause its out-neighbors to set their actual beliefs on $\theta^\star$ to be 0 by setting its own actual belief on $\theta^\star$ to be 0? We argue that this is impossible under the LFRHE algorithm. By way of contradiction, suppose there exists a time-step $\bar{t}(\omega)$ satisfying:

$$\bar{t}(\omega) = \min\{t \in \mathbb{N} : \exists i \in \mathcal{R} \text{ with } \mu_{i,t}(\theta^\star) = 0\}. \tag{5.58}$$

In words, $\bar{t}(\omega)$ represents the first time-step when some regular agent i sets its actual belief on the true hypothesis to be zero. Clearly, $\bar{t}(\omega) \neq 0$ based on line 1 of Algo. 4. Suppose $\bar{t}(\omega)$ is some positive integer, and focus on how agent i updates $\mu_{i,\bar{t}(\omega)}(\theta^\star)$ based on (5.13). Following similar arguments as in the proof of Lemma 5.8.2, we know that $\pi_{i,t}(\theta^\star) > 0, \forall t \in \mathbb{N}, \forall i \in \mathcal{R}$. At the same time, every belief featuring in the set $\Psi^{\theta^\star}_{i,\bar{t}(\omega)-1}$ (as defined in equation (5.57)) is strictly positive based on the way $\bar{t}(\omega)$ is defined. In light of the above arguments, and based on (5.56), (5.57), we infer:

$$\min\{\{\mu_{j,\bar{t}(\omega)-1}(\theta^\star)\}_{j \in \mathcal{M}^{\theta^\star}_{i,\bar{t}(\omega)-1}}, \pi_{i,\bar{t}(\omega)}(\theta^\star)\} > 0. \tag{5.59}$$

Thus, based on (5.13), we must have $\mu_{i,\bar{t}(\omega)}(\theta^\star) > 0$, yielding the desired contradiction. With $\eta(\omega) \triangleq \min\{\gamma_1 - \delta, \gamma_2(\omega)\} > 0$, one can easily verify the following by referring to (5.13):

$$\mu_{i,t}(\theta^\star) \geq \eta(\omega), \forall t \geq t'(\omega), \forall i \in \mathcal{R}. \tag{5.60}$$

In particular, (5.60) follows by (i) noting that for each $i \in \mathcal{R}$, $\pi_{i,t'(\omega)+1}(\theta^\star) \geq \eta(\omega)$, and each belief featuring in the set $\Psi^{\theta^\star}_{i,t'(\omega)}$ is lower bounded by $\eta(\omega)$, (ii) leveraging (5.56), (5.57), and (iii) using a similar string of arguments as those used to arrive at (5.25). Thus, we have established an analogous result as in Lemma 5.8.2 for the regular agents.

To proceed, let us fix a false hypothesis $\theta \neq \theta^\star$, and define

$$\tilde{K}(\theta^\star, \theta) \triangleq \min_{v \in \mathcal{S}(\theta^\star,\theta) \cap \mathcal{R}} K_v(\theta^\star, \theta)$$

. Then, given any $\epsilon > 0$, Lemma 5.8.1 implies the existence of a time-step $\tilde{t}_1(\omega, \theta, \epsilon)$, such that:

$$\pi_{i,t}(\theta) < e^{-(\tilde{K}(\theta^\star,\theta)-\epsilon)t}, \forall t \geq \tilde{t}_1(\omega, \theta, \epsilon), \forall i \in \mathcal{S}(\theta^\star, \theta) \cap \mathcal{R}. \tag{5.61}$$

Let $\tilde{t}_2 = \max\{t'(\omega), \tilde{t}_1(\omega, \theta, \epsilon)\}$, where we have suppressed the dependence of $\tilde{t}_2$ on ω, θ and ϵ. For any agent $i \in \mathcal{S}(\theta^\star, \theta) \cap \mathcal{R}$, observe that based on (5.56), (5.57) and (5.60),

$$\min\{\{\mu_{j,t}(\theta^\star)\}_{j \in \mathcal{M}^{\theta^\star}_{i,t}}, \pi_{i,t+1}(\theta^\star)\} \geq \eta(\omega), \forall t \geq \tilde{t}_2. \tag{5.62}$$

Combining the above with a similar line of argument as used to arrive at (5.28), we obtain:

$$\mu_{i,t}(\theta) < C_1(\omega) e^{-(\tilde{K}(\theta^\star,\theta)-\epsilon)t}, \forall t \geq \tilde{t}_2 + 1, \forall i \in \mathcal{S}(\theta^\star, \theta) \cap \mathcal{R}, \tag{5.63}$$

where $C_1(\omega) = \eta(\omega)^{-1}$. If $\mathcal{V} \setminus \mathcal{S}(\theta^\star, \theta)$ is empty, then we are essentially done. Else, define

$$\mathcal{L}_1(\theta^\star, \theta) \triangleq \{i \in \mathcal{V} \setminus \mathcal{S}(\theta^\star, \theta) : |\mathcal{N}_i \cap \mathcal{S}(\theta^\star, \theta)| \geq (2f + 1)\}. \tag{5.64}$$

Whenever $\mathcal{V} \setminus \mathcal{S}(\theta^\star, \theta)$ is non-empty, we claim that $\mathcal{L}_1(\theta^\star, \theta)$ (as defined above) is also non-empty based on the hypothesis of the theorem. To see this, note that if $\mathcal{L}_1(\theta^\star, \theta)$ is empty, then $\mathcal{C} = \mathcal{V} \setminus \mathcal{S}(\theta^\star, \theta)$ is not $(2f + 1)$-reachable, violating the fact that $\mathcal{G}$ is strongly $(2f + 1)$-robust w.r.t. $\mathcal{S}(\theta^\star, \theta)$. We claim that the following holds for each $i \in \mathcal{L}_1(\theta^\star, \theta) \cap \mathcal{R}$:

$$\min_{j \in \mathcal{M}^{\theta}_{i,t}} \mu_{j,t}(\theta) < C_1(\omega) e^{-(\tilde{K}(\theta^\star,\theta)-\epsilon)t}, \forall t \geq \tilde{t}_2 + 1. \tag{5.65}$$

To verify the above claim, pick any agent $i \in \mathcal{L}_1(\theta^\star, \theta) \cap \mathcal{R}$, and suppose $t \geq \tilde{t}_2 + 1$. When $|\mathcal{M}^{\theta}_{i,t} \cap \{\mathcal{S}(\theta^\star, \theta) \cap \mathcal{R}\}| > 0$, the claim follows immediately based on (5.63). Consider the case when $|\mathcal{M}^{\theta}_{i,t} \cap \{\mathcal{S}(\theta^\star, \theta) \cap \mathcal{R}\}| = 0$. Since $i \in \mathcal{L}_1(\theta^\star, \theta)$, it has at least $(2f + 1)$ neighbors in $\mathcal{S}(\theta^\star, \theta)$, out of which at least $f + 1$ are regular based on the f-locality of the adversarial model. Since the set $\mathcal{J}^{\theta}_{i,t}$ has cardinality f, it must then be that $|\mathcal{U}^{\theta}_{i,t} \cap \{\mathcal{S}(\theta^\star, \theta) \cap \mathcal{R}\}| > 0$. Let $u \in \mathcal{U}^{\theta}_{i,t} \cap \{\mathcal{S}(\theta^\star, \theta) \cap \mathcal{R}\}$. Based on the way $\mathcal{M}^{\theta}_{i,t}$ is defined, it must be that $\mu_{j,t}(\theta) \leq \mu_{u,t}(\theta) < C_1(\omega) e^{-(\tilde{K}(\theta^\star,\theta)-\epsilon)t}, \forall j \in \mathcal{M}^{\theta}_{i,t}$, where the last inequality follows from (5.63). This establishes our claim regarding

(5.65). Now consider the update of $\mu_{i,t+1}(\theta)$ based on (5.13), when $t \geq \tilde{t}_2 + 1$. In light of the above arguments, the numerator of the fraction on the R.H.S. of (5.13) is upper-bounded by $C_1(\omega)e^{-(\tilde{K}(\theta^{\star},\theta)-\epsilon)t}$, while the denominator is lower-bounded by $\eta(\omega)$. We conclude that for all $i \in \mathcal{L}_1(\theta^{\star},\theta) \cap \mathcal{R}$:

$$\mu_{i,t}(\theta) < (C_1(\omega))^2 C_2(\theta,\epsilon) e^{-(\tilde{K}(\theta^{\star},\theta)-\epsilon)t}, \forall t \geq \tilde{t}_2 + 2, \tag{5.66}$$

where $C_2(\theta,\epsilon) = e^{(\tilde{K}(\theta^{\star},\theta)-\epsilon)}$. With $\mathcal{L}_0(\theta^{\star},\theta) \triangleq \mathcal{S}(\theta^{\star},\theta)$, we recursively define the sets $\mathcal{L}_r(\theta^{\star},\theta), 1 \leq r \leq (n-1)$ as:

$$\mathcal{L}_r(\theta^{\star},\theta) \triangleq \{i \in \mathcal{V} \setminus \bigcup_{q=0}^{r-1} \mathcal{L}_q(\theta^{\star},\theta) : |\mathcal{N}_i \cap \{\bigcup_{q=0}^{r-1} \mathcal{L}_q(\theta^{\star},\theta)\}| \geq (2f+1)\}. \tag{5.67}$$

We claim that the following is true for all $i \in \mathcal{L}_r(\theta^{\star},\theta) \cap \mathcal{R}$:

$$\mu_{i,t}(\theta) < (C_1(\omega))^{r+1} (C_2(\theta,\epsilon))^r e^{-(\tilde{K}(\theta^{\star},\theta)-\epsilon)t}, \forall t \geq \tilde{t}_2 + (r+1). \tag{5.68}$$

To prove the claim, we proceed via induction on r. The base cases when $r \in \{0,1\}$ have already been established. Suppose equation (5.68) holds for all $r \in \{0,\ldots,m-1\}$, where $m \in \{2,\ldots,n-1\}$. The claim easily extends to the case when $r = m$ by noting that (i) $\mathcal{L}_m(\theta^{\star},\theta)$ is non-empty if $\mathcal{V} \setminus \{\bigcup_{q=0}^{(m-1)} \mathcal{L}_q(\theta^{\star},\theta)\}$ is non-empty (based on the hypothesis of the theorem), (ii) any agent $i \in \mathcal{L}_m(\theta^{\star},\theta) \cap \mathcal{R}$ has at least $(2f+1)$ neighbors in the set $\bigcup_{q=0}^{(m-1)} \mathcal{L}_q(\theta^{\star},\theta)$, of which at least $f+1$ are regular (based on the f-locality of the adversarial model), and (iii) using the induction hypothesis and arguments similar to those used to arrive at (5.66). We have thus verified the correctness of (5.68). Now taking the natural log on both sides of (5.68), dividing throughout by t, simplifying, and then taking the limit inferior on both sides of the resulting inequality immediately leads to (5.15). Finally, to complete the proof, it suffices to note that $\bigcup_{q=0}^{(n-1)} \mathcal{L}_q(\theta^{\star},\theta) = \mathcal{R}$. ■

6. DISTRIBUTED HYPOTHESIS TESTING WITH SPARSE AND QUANTIZED COMMUNICATION

In this chapter, we revisit the problem of distributed inference/hypothesis testing, and focus on scenarios where communication between agents is costly, and takes place over channels with finite bandwidth. To reduce the number of communication rounds, we develop a novel event-triggered distributed learning rule that is based on the principle of diffusing low beliefs on each false hypothesis. Building on this principle, we design a trigger condition under which an agent broadcasts only those components of its belief vector that have adequate innovation, to only those neighbors that require such information. We prove that our rule guarantees convergence to the true state exponentially fast almost surely despite sparse communication, and that it has the potential to significantly reduce information flow from uninformative agents to informative agents. Next, to deal with finite-precision communication channels, we propose a distributed learning rule that leverages the idea of adaptive quantization. We show that by sequentially refining the range of the quantizers, every agent can learn the truth exponentially fast almost surely, while using just 1 bit to encode its belief on each hypothesis. For both our proposed algorithms, we rigorously characterize the trade-offs between communication-efficiency and learning rate. By doing so, we identify sparse communication regimes, and quantizer precision levels, under which our rules recover the best known long-run learning rate for this problem.

6.1 Introduction

Over the last couple of decades, there has been a significant shift in the model of computation - driven in part by the nature of emerging applications, and partly due to concerns of reliability and scalability - from that of a single centralized computing node to parallel, distributed architectures comprising of several devices. Depending upon the context, these devices could be smart phones interacting with the cloud in a Federated Learning setup, or wearable devices, autonomous vehicles in a modern Internet of Things (IoT) network. Typically, the devices in the above applications - henceforth referred to as agents - run on limited battery power, and setting up communication links between such agents incurs significant latency. Thus, the need arises to reduce the number of communication rounds. Moreover, the communication links themselves have finite bandwidth, dictating the need to compress messages appropriately. In short, the *communication bottleneck* described above poses a major technical challenge. Our goal in this chapter is to take a step towards resolving this challenge for the canonical problem of distributed inference/hypothesis testing - a problem that we studied in Chapter 5 under the assumption of a perfect communication model. We briefly remind the reader of the problem setting.

Consider a network of agents, where each agent receives a stream of private signals sequentially over time. The observations of each agent are generated by a common underlying distribution, parameterized by an unknown static quantity which we call the *true state of the world*. The task of the agents is to collectively identify this unknown quantity from a finite family of hypotheses, while relying solely on local interactions. As we discussed in Chapter 5, the distributed inference/hypothesis testing problem enjoys a rich history [127,128,133,134,137,138,140,142,146,147,156], where a variety of techniques have been proposed over the years, with more recent efforts directed towards improving the convergence rate. These techniques can be broadly classified in terms of the mechanism used to aggregate data: while consensus-based linear [127,128,133,134] and log-linear [137,138,140,142,156] rules have been extensively studied, in Chapter 5 we introduced a min-protocol that leads to the best known (asymptotic) learning rate for this problem.

In general, for the problem described above, no one agent can eliminate every false hypothesis on its own to uniquely learn the true state. This leads to a fundamental tension: although communication is costly (due to battery power constraints) and imprecise (due to finite bandwidths), it is also necessary. *How should the agents interact to learn the true state despite sparse and imprecise communication?* At the moment, a theoretical understanding of this question is lacking in the literature on distributed hypothesis testing. In this context, our main contributions are as follows.

6.1.1 Summary of Contributions

To reduce the number of communication rounds, one needs to first answer a few basic questions. (i) When should an agent exchange information with a neighbor? (ii) What piece of information should the agent exchange? To address these questions in a principled way, our first contribution is to develop a novel distributed learning rule in Section 6.3 by drawing on ideas from event-triggered control [157, 158]. The premise of our rule is based on diffusing low beliefs on each false hypothesis across the network. Building on this principle, we design a trigger condition that carefully takes into account the specific structure of the problem, and enables an agent to decide, using purely local information, whether or not to broadcast its belief[1] on a given hypothesis to a given neighbor. Specifically, based on our event-triggered strategy, an agent broadcasts *only* those components of its belief vector that have adequate "innovation", to *only* those neighbors that are in need of the corresponding pieces of information. Thus, our approach not only reduces the number of communication rounds, but also the amount of information transmitted in each round.

Our second contribution is to provide a detailed theoretical characterization of the proposed event-triggered learning rule in Section 6.4. Specifically, in Theorem 6.4.1 we establish that our rule enables each agent to learn the true state exponentially fast almost surely, under standard assumptions on the observation model and the network topology. We characterize the learning rate of our algorithm as a function of the agents' relative entropies, the network structure, and parameters of the communication model. In particular, we show that even when the inter-communication

[1] By an agent's belief vector, we imply a distribution over the set of hypotheses; this vector gets recursively updated over time as an agent acquires more information.

intervals between the agents grow geometrically at a rate $p > 1$, our rule guarantees exponentially fast learning at a network-dependent rate that scales inversely with p. However, when such intervals grow polynomially, the learning rate remains the same as the best known network-independent learning rate of [147]. Thus, our results provide various interesting insights into the relationship that exists between the rate of convergence and the sparsity of the communication pattern.

Next, in Propositions 6.4.1 and 6.4.2, we demonstrate that our event-triggered scheme has the potential to significantly reduce information flow from uninformative agents to informative agents. Finally, in Theorem 6.4.3, we argue that if asymptotic learning of the true state is the only consideration, then one can allow for communication schemes with arbitrarily long intervals between successive communications.

While our results above concern the aspect of sparse communication, in Section 6.5 we turn our attention to learning over communication channels with finite precision, i.e., channels that can support only a finite number of bits. In a recent paper [140] that looks at the same problem as us, the authors demonstrated in simulations that with a quantized variant of their log-linear rule, the beliefs of the agents might converge to a wrong hypothesis, if not enough bits are used to encode the beliefs. This raises the following fundamental question. *In order to learn the true state, how many bits must an agent use to encode its belief on each hypothesis?* To answer this question, we develop a distributed learning rule based on the idea of adaptive quantization. The key feature of our rule is to successively refine the range of the quantizers as the agents acquire more information over time and narrow down on the truth. In Theorem 6.6.1, we prove that even if every agent uses just 1 bit to encode its belief on each hypothesis, all agents end up learning the truth exponentially fast almost surely. The rate of learning, however, exhibits a dependence on the precision of the quantizer - a dependence that we explicitly characterize. In doing so, we show that if the number of bits used for encoding each hypothesis is chosen to be large enough w.r.t. certain relative entropies, then one can recover the exact same long-run learning rate as with infinite precision, i.e., the rate obtained in [147]. This constitutes our final contribution.

To summarize, this chapter (i) develops novel communication-efficient distributed learning algorithms; (ii) provides detailed theoretical characterizations of their performance; and, in particular, (iii) highlights various interesting trade-offs between sparse and imprecise communication, and the learning rate.

A preliminary version of the results in this chapter appeared as [159], and is available in part in the pre-print [160].

6.1.2 Related Work

Our work is closely related to the papers [161] and [162], each of which explores the theme of event-driven communications for distributed learning. In [161], the authors propose a rule where an agent queries the log-marginals of its neighbors only if the total variation distance between its current belief and the Bayesian posterior after observing a new signal falls below a pre-defined threshold. That is, an agent communicates only if its current private signal is not adequately informative. Among various other differences, the trigger condition we propose is not only a function of an agent's local observations, but also carefully incorporates feedback from neighboring agents. Moreover, while we provide theoretical results to substantiate that our rule leads to sparse communication patterns, [161] does so only via simulations. The algorithm in [162] comes with no theoretical guarantees of convergence.

The aspect of sparse communication has been studied in the context of a variety of coordination problems on networks, such as average consensus [163], distributed optimization [164, 165], and static parameter estimation [166] - settings that differ from the one we investigate in this chapter. To promote communication-efficiency, [164] and [166] propose algorithms where inter-agent interactions become progressively sparser over time. However, these algorithms are essentially time-triggered, i.e., they do not adhere to the principle that "an agent should communicate only when it has something useful to say". On the other hand, the strand of literature that deals with event-driven communications for multi-agent systems focuses primarily on variations of the basic consensus problem; we refer the reader to [167] for a survey of such techniques. Notably, the common recipe for designing such techniques centers around a Lyapunov argument for deterministic systems. However, it is not at all apparent how such design ideas can be exploited for the stochastic inference problem we consider here.[2]

Our work is also related to the classical literature on decentralized hypothesis testing under communication constraints [168–170]. However, unlike our formulation, these papers assume the presence of a centralized fusion center, and do not deal with sequential data, i.e., each agent only receives one signal. Finally, we point out that the adaptive quantization idea employed in this paper bears conceptual similarities to the encoding-decoding strategy in [171] for stabilizing an LTI plant over a bit-constrained channel, and also to a recent work on distributed optimization [172].

6.2 Model

The network and observation models that we discuss next are essentially the same as those in Section 5.2 of Chapter 5. We repeat them here to keep the chapter self-contained.

Network Model: We consider a group of agents $\mathcal{V} = \{1, \ldots, n\}$, and model interactions among them via an undirected graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$.[3] An edge $(i, j) \in \mathcal{E}$ indicates that agent i can directly transmit information to agent j, and vice versa. The set of all neighbors of agent i is defined as $\mathcal{N}_i = \{j \in \mathcal{V} : (j, i) \in \mathcal{E}\}$. We say that $\mathcal{G}$ is rooted at $\mathcal{C} \subseteq \mathcal{V}$, if for each agent $i \in \mathcal{V} \setminus \mathcal{C}$, there exists a path to it from some agent $j \in \mathcal{C}$. For a connected graph $\mathcal{G}$, we will use $d(i, j)$ to denote the length of the shortest path between i and j.

[2] The stochastic nature of our problem arises from the fact that the signals seen by each agent are random variables.

[3] The results in this chapter can be easily extended to directed graphs.

Observation Model: Let $\Theta = \{\theta_1, \theta_2, \ldots, \theta_m\}$ denote m possible states of the world, with each state representing a hypothesis. A specific state $\theta^\star \in \Theta$, referred to as the true state of the world, gets realized. Conditional on its realization, at each time-step $t \in \mathbb{N}_+$, every agent $i \in \mathcal{V}$ privately observes a signal $s_{i,t} \in \mathcal{S}_i$, where $\mathcal{S}_i$ denotes the signal space of agent i. The joint observation profile so generated across the network is denoted $s_t = (s_{1,t}, s_{2,t}, \ldots, s_{n,t})$, where $s_t \in \mathcal{S}$, and $\mathcal{S} = \mathcal{S}_1 \times \mathcal{S}_2 \times \ldots \mathcal{S}_n$. Specifically, the signal s_t is generated based on a conditional likelihood function $l(\cdot|\theta^\star)$, the i-th marginal of which is denoted $l_i(\cdot|\theta^\star)$, and is available to agent i. The signal structure of each agent $i \in \mathcal{V}$ is thus characterized by a family of parameterized marginals $l_i = \{l_i(w_i|\theta) : \theta \in \Theta, w_i \in \mathcal{S}_i\}$. We make the same asssumptions on the observation model as in Chapter 5: (i) The signal space of each agent i, namely $\mathcal{S}_i$, is finite. (ii) Each agent i has knowledge of its local likelihood functions $\{l_i(\cdot|\theta_p)\}_{p=1}^{m}$, and it holds that $l_i(w_i|\theta) > 0, \forall w_i \in \mathcal{S}_i$, and $\forall \theta \in \Theta$. (iii) The observation sequence of each agent is described by an i.i.d. random process over time; however, at any given time-step, the observations of different agents may potentially be correlated. (iv) There exists a fixed true state of the world $\theta^\star \in \Theta$ (unknown to the agents) that generates the observations of all the agents. The probability space for our model is denoted $(\Omega, \mathcal{F}, \mathbb{P}^{\theta^\star})$, where $\Omega \triangleq \{\omega : \omega = (s_1, s_2, \ldots), \forall s_t \in \mathcal{S}, \forall t \in \mathbb{N}_+\}$, $\mathcal{F}$ is the σ-algebra generated by the observation profiles, and $\mathbb{P}^{\theta^\star}$ is the probability measure induced by sample paths in Ω. Specifically, $\mathbb{P}^{\theta^\star} = \prod_{t=1}^{\infty} l(\cdot|\theta^\star)$. We will use the abbreviation a.s. to indicate almost sure occurrence of an event w.r.t. $\mathbb{P}^{\theta^\star}$.

The goal of each agent in the network is to eventually learn the true state $\theta^\star$. As we pointed out in Chapter 5, this task is complicated by the fact that for any given agent i, certain states might be *observationally equivalent* to θ^* from its perspective. Essentially, what this means is that by inspecting its pattern of private signals, agent i may not be able to uniquely identify the truth. Our broad **goal** in this chapter is to develop distributed learning algorithms that not only resolve this identifiability problem, but do so in the face of *sparse* and *imprecise* communication. To this end, we recall the following definition from Chapter 5.

Definition 6.2.1 ***(Source agents)*** *An agent i is said to be a source agent for a pair of distinct hypotheses $\theta_p, \theta_q \in \Theta$ if it can distinguish between them, i.e., if $K_i(\theta_p, \theta_q) > 0$, where $K_i(\theta_p, \theta_q)$ represents the KL-divergence [149] between the distributions $l_i(\cdot|\theta_p)$ and $l_i(\cdot|\theta_q)$. The set of source agents for pair (θ_p, θ_q) is denoted $\mathcal{S}(\theta_p, \theta_q)$.*

6.3 An Event-Triggered Distributed Learning Rule

• **Belief-Update Strategy**: In this section, we develop an event-triggered distributed learning rule that enables each agent to eventually learn the truth, despite infrequent information exchanges with its neighbors. Our approach requires each agent i to maintain a local belief vector $\boldsymbol{\pi}_{i,t}$, and an actual belief vector $\boldsymbol{\mu}_{i,t}$, each of which are probability distributions over the hypothesis set Θ. While agent i updates $\boldsymbol{\pi}_{i,t}$ in a Bayesian manner using only its private signals (see eq. (6.2)), to formally describe how it updates $\boldsymbol{\mu}_{i,t}$, we need to first introduce some notation. Accordingly, let $\mathbb{1}_{ji,t}(\theta) \in \{0, 1\}$ be an indicator variable which takes on a value of 1 if and only if agent j broadcasts $\mu_{j,t}(\theta)$ to agent i at time t. Next, we define $\mathcal{N}_{i,t}(\theta) \triangleq \{j \in \mathcal{N}_i | \mathbb{1}_{ji,t}(\theta) = 1\}$ as the subset of agent i's neighbors who broadcast their belief on θ to i at time t. As part of our learning algorithm, each agent i keeps track of the lowest belief on each hypothesis $\theta \in \Theta$ that it has heard up to any given instant t, denoted by $\bar{\mu}_{i,t}(\theta)$. More precisely, $\bar{\mu}_{i,0}(\theta) = \mu_{i,0}(\theta)$, and $\forall t+1 \in \mathbb{N}_+$,

$$\bar{\mu}_{i,t+1}(\theta) = \min\{\bar{\mu}_{i,t}(\theta), \{\mu_{j,t+1}(\theta)\}_{j \in \{i\} \cup \mathcal{N}_{i,t+1}(\theta)}\}. \tag{6.1}$$

We are now in position to describe the belief-update rule at each agent: $\boldsymbol{\pi}_{i,t}$ and $\boldsymbol{\mu}_{i,t}$ are initialized with $\pi_{i,0}(\theta) > 0, \mu_{i,0}(\theta) > 0, \forall \theta \in \Theta, \forall i \in \mathcal{V}$ (but otherwise arbitrarily), and subsequently updated as follows $\forall t+1 \in \mathbb{N}_+$:

$$\pi_{i,t+1}(\theta) = \frac{l_i(s_{i,t+1}|\theta)\pi_{i,t}(\theta)}{\sum\limits_{p=1}^{m} l_i(s_{i,t+1}|\theta_p)\pi_{i,t}(\theta_p)}, \tag{6.2}$$

$$\mu_{i,t+1}(\theta) = \frac{\min\{\bar{\mu}_{i,t}(\theta), \pi_{i,t+1}(\theta)\}}{\sum\limits_{p=1}^{m} \min\{\bar{\mu}_{i,t}(\theta_p), \pi_{i,t+1}(\theta_p)\}}. \tag{6.3}$$

• **Communication Strategy**: We now focus on specifying *when* an agent broadcasts its belief on a given hypothesis to a neighbor. To this end, we first define a sequence $\mathbb{I} = \{t_k\}$ of *event-monitoring* time-steps, where $t_1 = 1$, and $t_{k+1} - t_k = g(k), \forall k \in \mathbb{N}_+$. Here, $g : \mathbb{R}_+ \to \mathbb{R}_+$ is a continuous, non-decreasing function that takes on integer values at integers. We will henceforth refer to $g(k)$ as the *event-interval* function. At any given time $t \in \mathbb{N}_+$, let $\hat{\mu}_{ij,t}(\theta)$ represent agent i's belief on θ the last time (excluding time t) it transmitted its belief on θ to agent j. Our communication strategy is as follows. At t_1, each agent $i \in \mathcal{V}$ broadcasts its entire belief vector $\boldsymbol{\mu}_{i,t}$ to every neighbor. Subsequently, at each $t_k, k \geq 2$, i transmits $\mu_{i,t_k}(\theta)$ to $j \in \mathcal{N}_i$ if and only if the following event occurs:

$$\mu_{i,t_k}(\theta) < \gamma(t_k) \min\{\hat{\mu}_{ij,t_k}(\theta), \hat{\mu}_{ji,t_k}(\theta)\}, \tag{6.4}$$

where $\gamma : \mathbb{N} \to (0, 1]$ is a non-increasing function, which we will henceforth call the *threshold* function. If $t \notin \mathbb{I}$, then an agent i does not communicate with its neighbors at time t, i.e., all inter-agent interactions are restricted to time-steps in $\mathbb{I}$, subject to the trigger-condition given by (6.4). Notice that we have not yet specified the functional forms of $g(\cdot)$ and $\gamma(\cdot)$; we will comment on this topic later in Section 6.4.

• **Summary**: At each time-step $t + 1 \in \mathbb{N}_+$, and for each hypothesis $\theta \in \Theta$, the sequence of operations executed by an agent i is summarized as follows. (i) Agent i updates its local and actual beliefs on θ via (6.2) and (6.3), respectively. (ii) For each neighbor $j \in \mathcal{N}_i$, it decides whether or not to transmit $\mu_{i,t+1}(\theta)$ to j, and collects $\{\mu_{j,t+1}(\theta)\}_{j \in \mathcal{N}_{i,t+1}(\theta)}$.[4] (iii) It updates $\bar{\mu}_{i,t+1}(\theta)$ via (6.1) using the (potentially) new information it acquires from its neighbors at time $t + 1$. We call the above algorithm the `Event-Triggered Min-Rule` and outline its steps in Algorithm 5.

• **Intuition:** The premise of our belief-update strategy is based on diffusing low beliefs on each false hypothesis. For a given false hypothesis θ, the local Bayesian update (6.2) will generate a decaying sequence $\pi_{i,t}(\theta)$ for each $i \in \mathcal{S}(\theta^*, \theta)$. Update rules (6.1) and (6.3) then help propagate agent i's low belief on θ to the rest of the

[4] If $t + 1 \notin \mathbb{I}$, this step gets bypassed, and $\mathcal{N}_{i,t+1}(\theta) = \emptyset, \forall \theta \in \Theta$.

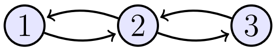


Fig. 6.1. The figure shows a network where only agent 1 is informative. In Section 6.3, we design an event-triggered algorithm under which all upstream broadcasts along the path 3 → 2 → 1 stop eventually almost surely. At the same time, all agents learn the true state. We demonstrate these facts both in theory (see Sec. 6.4), and in simulations (see Sec. 6.7).

Algorithm 5 (Event-Triggered Min-Rule) Each agent $i \in \mathcal{V}$ executes this algorithm in parallel

Initialization: $\mu_{i,0}(\theta) > 0$, $\pi_{i,0}(\theta) > 0$, $\bar{\mu}_{i,0}(\theta) = \mu_{i,0}(\theta), \forall \theta \in \Theta$, and $\sum_{\theta \in \Theta} \mu_{i,0}(\theta) = 1$, $\sum_{\theta \in \Theta} \pi_{i,0}(\theta) = 1$.

```
for $t+1 \in \mathbb{N}_+$ do
    for $\theta \in \Theta$ do
        Update $\pi_{i,t+1}(\theta)$ via (6.2), and $\mu_{i,t+1}(\theta)$ via (6.3).
        if $t+1 = t_1$ then
            Broadcast $\mu_{i,t+1}(\theta)$ to each $j \in \mathcal{N}_i$.
        else
            For each $j \in \mathcal{N}_i$, broadcast $\mu_{i,t+1}(\theta)$ to $j$ if and only if $t+1 \in \mathbb{I}$ *and* the event condition (6.4) holds.
        end if
        Receive $\mu_{j,t+1}(\theta)$ from each $j \in \mathcal{N}_{i,t+1}(\theta)$, and update $\bar{\mu}_{i,t+1}(\theta)$ via (6.1).
    end for
end for
```

network. We point out that in contrast to the min-rule that we developed in Chapter 5, where for updating $\mu_{i,t+1}(\theta)$, agent i used the lowest neighboring belief on θ at the *previous* time-step t, our approach here requires an agent i to use the lowest belief on θ that it has heard *up to* time t, namely $\bar{\mu}_{i,t}(\theta)$. This modification will be crucial in the convergence analysis of Algorithm 5.

To build intuition regarding our communication strategy, let us consider the network in Fig 6.1. Suppose $\Theta = \{\theta_1, \theta_2\}, \theta^* = \theta_1$, and $\mathcal{S}(\theta_1, \theta_2) = 1$, i.e., agent 1 is the only informative agent. Since our principle of learning is based on eliminating each false hypothesis, it makes sense to broadcast beliefs only if they are low enough. Based on this observation, one naive approach to enforce sparse communication could be to set a fixed low threshold, say β, and wait till beliefs fall below such a threshold to broadcast. While this might lead to sparse communication initially, in order to learn the truth, there must come a time beyond which the beliefs of all agents on the false hypothesis θ_2 always stay below β, leading to dense communication eventually. The obvious fix is to introduce an event-condition that is *state-dependent*. Consider the following candidate strategy: an agent broadcasts its belief on a state θ only if it is sufficiently lower than what it was when it last broadcasted about θ. While an improvement over the "fixed-threshold" strategy, this new scheme has the following demerit: broadcasts are not *agent-specific*. In other words, going back to our example, agent 2 (resp., agent 3) might transmit unsolicited information to agent 1 (resp., agent 2) - information, that agent 1 (resp., agent 2) can do without. To remedy this, one can consider a request/poll based scheme as in [161] and [173], where an agent receives information from a neighbor only by polling that neighbor. However, now each time agent 2 needs information from agent 1, it needs to place a request, *the request itself incurring extra communication.*

Given the above issues, we ask: Is it possible to devise an event-triggered scheme that eventually stops unnecessary broadcasts from agents 3 to 2, and 2 to 1, while preserving essential information flow from agents 1 to 2, and 2 to 3? More generally, we seek a triggering rule that can reduce transmissions from uninformative agents to informative agents. This leads us to the event condition in Eq. 6.4. For each $\theta \in \Theta$, an agent i broadcasts $\mu_{i,t}(\theta)$ to a neighbor $j \in \mathcal{N}_i$ only if $\mu_{i,t}(\theta)$ has adequate "innovation" w.r.t. i's last broadcast about θ to j, *and* j's last broadcast about θ to i. A decreasing threshold function $\gamma(t)$ makes it progressively harder to satisfy the event condition in Eq. 6.4, demanding more innovation to merit broadcast as time progresses.[5] The rationale behind checking the event condition only at time-steps in $\mathbb{I}$ is twofold.[6] First, it saves computations since the event condition need not be checked all the time. Second, and more importantly, it provides an additional instrument to control communication-sparsity on top of event-triggering. Indeed, a monotonically increasing event-interval function $g(\cdot)$ implies fewer agent interactions with time, since all potential broadcasts are restricted to $\mathbb{I}$. In particular, without the event condition in Eq. 6.4, our communication strategy would boil down to a simple time-triggered rule, akin to the one studied in our recent work [159].

We close this section by highlighting that our event condition (i) is *θ-specific*, since an agent may not be equally informative about all states[7]; (ii) is *neighbor-specific*, since not all neighbors might require information; (iii) is *problem-specific*, since it is built upon the principle of eliminating false hypotheses by diffusing low beliefs; and (iv) can be checked using local information only.

6.4 Theoretical Guarantees for Algorithm 5

In this section, we state the main results pertaining to our `Event-Triggered Min-Rule`, and then discuss their implications. Proofs of these results are deferred to Section 6.9.1. To state the first result concerning the convergence of our learning rule, let $G(\cdot)$ be used to denote the integral of $g(\cdot)$, and $G^{-1}(\cdot)$ represent the inverse of $G(\cdot)$. Since $g(\cdot)$ is strictly positive by definition, $G(\cdot)$ is strictly increasing, and hence, $G^{-1}(\cdot)$ is well-defined.

Theorem 6.4.1 *Suppose the functions $g(\cdot)$ and $\gamma(\cdot)$ satisfy:*

$$\lim_{t\to\infty} \frac{G(G^{-1}(t)-2)}{t} = \alpha \in (0,1]; \quad \lim_{t\to\infty} \frac{\log(1/\gamma(t))}{t} = 0. \tag{6.5}$$

[5] We will see later on (Prop. 6.4.2) that for the network in Fig. 6.1, this scheme provably stops communications from agents 3 to 2, and 2 to 1, eventually.

[6] While this might appear similar to the Periodic Event-Triggering (PETM) framework [174] where events are checked periodically, the sequence $\mathbb{I}$ can be significantly more general than a simple periodic sequence.

[7] This is precisely the motivation behind tracking changes in individual components of the belief vector, as opposed to looking at changes in the overall belief vector using, for instance, the total variation metric.

Furthermore, suppose the following conditions hold. (i) For every pair of hypotheses $\theta_p, \theta_q \in \Theta$, the source set $\mathcal{S}(\theta_p, \theta_q)$ is non-empty. (ii) The communication graph $\mathcal{G}$ is connected. Then, Algorithm 5 guarantees the following.

- ***(Consistency)****: For each agent $i \in \mathcal{V}$, $\mu_{i,t}(\theta^\star) \rightarrow 1$ a.s.*
- ***(Exponentially Fast Rejection of False Hypotheses)****: For each agent $i \in \mathcal{V}$, and for each false hypothesis $\theta \in \Theta \setminus \{\theta^\star\}$, the following holds:*

$$\liminf_{t\to\infty} -\frac{\log \mu_{i,t}(\theta)}{t} \geq \max_{v \in \mathcal{S}(\theta^\star,\theta)} \alpha^{d(v,i)} K_v(\theta^\star, \theta) \; a.s. \tag{6.6}$$

At this point, it is natural to ask: For what classes of functions $g(\cdot)$ does the result of Theorem 6.4.1 hold? The following result provides an answer.

Corollary 6.4.2 *Suppose the conditions in Theorem 6.4.1 hold.*

(i) Suppose $g(x) = x^p, \forall x \in \mathbb{R}_+$, where p is any positive integer. Then, for each $\theta \in \Theta \setminus \{\theta^\star\}$, and $i \in \mathcal{V}$:

$$\liminf_{t\to\infty} -\frac{\log \mu_{i,t}(\theta)}{t} \geq \max_{v \in \mathcal{S}(\theta^\star,\theta)} K_v(\theta^\star, \theta) \; a.s. \tag{6.7}$$

(ii) Suppose $g(x) = p^x, \forall x \in \mathbb{R}_+$, where p is any positive integer. Then, for each $\theta \in \Theta \setminus \{\theta^\star\}$, and $i \in \mathcal{V}$:

$$\liminf_{t\to\infty} -\frac{\log \mu_{i,t}(\theta)}{t} \geq \max_{v \in \mathcal{S}(\theta^\star,\theta)} \frac{K_v(\theta^\star, \theta)}{p^{2d(v,i)}} \; a.s. \tag{6.8}$$

Proof The proof follows by directly computing the limit in Eq. (6.5). For case (i), $\alpha = 1$, and for case (ii), $\alpha = 1/p^2$. ■

Clearly, based on the rules of Algorithm 5, the communication pattern between the agents is at least as sparse as the sequence $\mathbb{I}$. The event-triggering strategy that we employ introduces further sparsity, as we establish in the next result.

Proposition 6.4.1 *Suppose the conditions in Theorem 6.4.1 are met. Then, there exists $\bar{\Omega} \subseteq \Omega$ such that $\mathbb{P}^{\theta^*}(\bar{\Omega}) = 1$, and for each $\omega \in \bar{\Omega}$, $\exists T_1(\omega), T_2(\omega) < \infty$ such that the following hold.*

(i) At each $t_k \in \mathbb{I}$ such that $t_k > T_1(\omega)$, $\mathbb{1}_{ij,t_k}(\theta^*) \neq 1, \forall i \in \mathcal{V}$ and $\forall j \in \mathcal{N}_i$.

(ii) Consider any $\theta \neq \theta^*$, and $i \notin \mathcal{S}(\theta^*, \theta)$. Then, at each $t_k > T_2(\omega)$, $\exists j \in \mathcal{N}_i$ such that $\mathbb{1}_{ij,t_k}(\theta) \neq 1$.[8]

The following result is an immediate application of the above proposition.

Proposition 6.4.2 *Suppose the conditions in Theorem 6.4.1 are met. Additionally, suppose $\mathcal{G}$ is a tree graph, and for each pair $\theta_p, \theta_q \in \Theta$, $|\mathcal{S}(\theta_p, \theta_q)| = 1$. Consider any $\theta \neq \theta^*$, and let $\mathcal{S}(\theta^*, \theta) = v_\theta$. Then, each agent $i \in \mathcal{V} \setminus \{v_\theta\}$ stops broadcasting its belief on θ to its parent in the tree rooted at v_θ eventually almost surely.*

A few comments are now in order.

- **On the nature of $g(\cdot)$ and $\gamma(\cdot)$:** Intuitively, if the event-interval function $g(\cdot)$ does not grow too fast, and the threshold function $\gamma(\cdot)$ does not decay too fast, one should expect things to fall in place. Theorem 6.4.1 makes this intuition precise by identifying conditions on $g(\cdot)$ and $\gamma(\cdot)$ that lead to exponentially fast learning of the truth. In particular, our framework allows for a considerable degree of freedom in the choice of $g(\cdot)$ and $\gamma(\cdot)$. Indeed, from (6.5), we note that any $\gamma(\cdot)$ that decays sub-exponentially works for our purpose. Moreover, Corollary 6.4.2 reveals that up to integer constraints, $g(\cdot)$ can be any polynomial or exponential function.

- **Trade-offs between sparse communication and learning rate:** What is the price paid for sparse communication? To answer the above question, we set as benchmark the scenario studied in our previous work [147], where we did not account for communication efficiency. There, we showed that each false hypothesis θ gets rejected exponentially fast by every agent at the *network-independent* rate

[8] In this claim, j might depend on t_k.

$\max_{v\in\mathcal{V}} K_v(\theta^*,\theta)$ - the *best* known rate in the existing literature on this problem.[9] From (6.6), we note that under highly sparse communication regimes which correspond to $\alpha < 1$, although learning occurs exponentially fast, the learning rate gets lowered relative to [147]. Moreover, unlike [147], (6.6) reveals that the asymptotic learning rate is *network-dependent* and *agent-specific*, i.e., different agents may discover the truth at different rates. In particular, when considering the asymptotic rate of rejection of a particular false hypothesis at a given agent i, notice from the R.H.S. of (6.6) that one needs to account for the attenuated relative entropies of the corresponding source agents, where the attenuation factor scales exponentially with the distances of agent i from such source agents. An instance of the above scenario is when the inter-communication intervals grow geometrically at rate $p > 1$; see case (ii) of Corollary 6.4.2.

On the other hand, from case (i) of Corollary 6.4.2, we glean that, polynomially growing inter-communication intervals, coupled with our proposed event-triggering strategy, lead to *no loss in the long-term learning rate relative to the benchmark case in* [147], i.e., as far as asymptotic performance is concerned, communication-efficiency comes essentially for "free" under this regime.

• **Sparse communication introduced by event-triggering:** Observe that being able to eliminate each false hypothesis is enough for learning the true state. In other words, agents need not exchange their beliefs on the true state (of course, no agent knows a priori what the true state is). Our event-triggering scheme precisely achieves this, as evidenced by claim (i) of Proposition 6.4.1: every agent stops broadcasting its belief on θ^* eventually almost surely. In addition, an important property of our event-triggering strategy is that it reduces information flow from uninformative agents to informative agents. To see this, consider any false hypothesis $\theta \neq \theta^*$, and an agent $i \notin \mathcal{S}(\theta^*,\theta)$. Since $i \notin \mathcal{S}(\theta^*,\theta)$, agent i's local belief $\pi_{i,t}(\theta)$ will stop decaying eventually, making it impossible for agent i to lower its actual belief $\mu_{i,t}(\theta)$ without

[9] For linear [127,128,133,134] and log-linear [137,138,140,142,156] learning rules, the corresponding rate is a convex combination of the relative entropies $K_v(\theta^*,\theta), v \in \mathcal{V}$.

the influence of its neighbors. Consequently, when left alone between consecutive event-monitoring time-steps, i will not be able to leverage its own private signals to generate enough "innovation" in $\mu_{i,t}(\theta)$ to broadcast to the neighbor who most recently contributed to lowering $\mu_{i,t}(\theta)$. The intuition here is simple: an uninformative agent cannot outdo the source of its information. This idea is made precise in claim (ii) of Proposition 6.4.1. To further demonstrate this facet of our rule, Proposition 6.4.2 stipulates that when the baseline graph is a tree, then all upstream broadcasts to informative agents stop after a finite period of time.

6.4.1 Asymptotic Learning of the Truth

If asymptotic learning of the true state is all one cares about, i.e., if exponential convergence is no longer a consideration, then one can allow for arbitrarily sparse communication patterns, as we shall now demonstrate. Accordingly, we first allow the baseline graph $\mathcal{G}(t) = (\mathcal{V}, \mathcal{E}(t))$ to now change over time. To allow for this generality, we set $\mathbb{I} = \mathbb{N}_+$, i.e., the event condition (6.4) is now monitored at each time-step. Furthermore, we set $\gamma(t) = \gamma \in (0, 1], \forall t \in \mathbb{N}$. At each time-step $t \in \mathbb{N}_+$, and for each $\theta \in \Theta$, an agent $i \in \mathcal{V}$ decides whether or not to broadcast $\mu_{i,t}(\theta)$ to an instantaneous neighbor $j \in \mathcal{N}_i(t)$ by checking the event condition (6.4). While checking this condition, if agent i has not yet transmitted to (resp., heard from) agent j about θ prior to time t, then it sets $\hat{\mu}_{ij,t}(\theta)$ (resp., $\hat{\mu}_{ji,t}(\theta)$) to 1. Update rules (6.1), (6.2), (6.3) remain the same, with $\mathcal{N}_{i,t}(\theta)$ now interpreted as $\mathcal{N}_{i,t}(\theta) \triangleq \{j \in \mathcal{N}_i(t) | \mathbb{1}_{ji,t}(\theta) = 1\}$. Finally, by an union graph over an interval $[t_1, t_2]$, we will imply the graph with vertex set $\mathcal{V}$, and edge set $\cup_{\tau=t_1}^{t_2} \mathcal{E}(\tau)$. With these modifications in place, we have the following result.

Theorem 6.4.3 *Suppose for every pair of hypotheses $\theta_p, \theta_q \in \Theta$, $\mathcal{S}(\theta_p, \theta_q)$ is non-empty. Furthermore, suppose for each $t \in \mathbb{N}_+$, the union graph over $[t, \infty)$ is rooted at $\mathcal{S}(\theta_p, \theta_q)$. Then, the event-triggered distributed learning rule described above guarantees $\mu_{i,t}(\theta^*) \to 1$ a.s. $\forall i \in \mathcal{V}$.*

While a result of the above flavor is well known for the basic consensus setting [175], we are unaware of its analogue for the distributed inference problem. When $\mathcal{G}(t) = \mathcal{G}, \forall t \in \mathbb{N}$, we observe from Theorem 6.4.3 that, as long as each agent transmits its belief vector to every neighbor infinitely often, all agents will asymptotically learn the truth. In particular, other than the above requirement, our result places no constraints on the *frequency* of agent interactions.

6.5 A Distributed Learning Rule based on Adaptive Quantization

The focus of Section 6.3 was on designing an algorithm that guarantees learning despite sparse communication. In this section, we turn our attention to promoting communication-efficiency via a complementary mechanism, namely, by compressing the amount of information transmitted by each agent. Our investigations here are motivated by the fact that in practice, communication channels modeling the interactions between agents have finite bandwidth. Accordingly, let us suppose that each agent i uses only $B(\theta)$ bits to encode its belief on θ. Under what conditions on $B(\theta)$ will each agent eventually learn the true state?

To answer the above question, we need to design an appropriate quantization scheme, which, in turn, requires resolving the following issues. (1) The scheme should be such that the belief of each agent on θ^* converges *exactly* to 1, as opposed to getting stuck in a neighborhood of 1. There are in fact various examples in the literature where due to quantization effects, the iterates of the algorithm converge to a neighborhood of the desired point [176–178]. (2) Precaution needs to be taken to ensure that the belief of an agent on θ^* never gets quantized to 0. Indeed, it might very well be that during an initial transient phase, the belief of some agent on θ^* falls inadvertently. If the quantization scheme is not designed appropriately, such a low belief on θ^* might get quantized to a 0 value, causing every agent to eventually place a 0 belief on the true state. This is a serious issue that needs to be addressed, and, in fact, this exact phenomenon has been reported in a simulation study conducted in [140]. Specifically, the authors in [140] present an example where using 12 bits to represent each hypothesis leads to learning the true state, but using 8 bits results in convergence to a false hypothesis. In what follows, we propose an algorithm that tackles the above issues; later, we argue that our algorithm guarantees exponentially fast learning even when merely 1 bit is used to encode each hypothesis.

To proceed, suppose we wish to encode a scalar x that belongs to the interval $[L, U]$ using B bit precision. Then, we first divide the interval $[L, U]$ into 2^B bins, each of equal width. Next, we identify the bin to which x belongs, and let the quantized value of x simply be the upper end point of that bin. Let this entire operation be described formally by a map $\mathcal{Q}_{R,B}(\cdot)$ with range parameter $R = [L, U]$ and bit parameter B. Then, we have $\mathcal{Q}_{R,B}(x) = L + d\lceil (x - L)/d \rceil$, where $d = (U - L)/2^B$. The above encoder will serve as a basic building block for encoding each component of an agent's belief vector, and our key idea will be to sequentially refine the range of the quantizer as more information is acquired over time.

- **Encoding Beliefs:** As with Algorithm 5, each agent i maintains a local belief vector $\boldsymbol{\pi}_{i,t}$, and an actual belief vector $\boldsymbol{\mu}_{i,t}$, which are updated via (6.2) and (6.3), respectively. In addition, for encoding its belief on θ, an agent i maintains a quantity $q_{i,t}(\theta)$, with $q_{i,0}(\theta) = 1, \forall \theta \in \Theta$. At each time-step $t + 1 \in \mathbb{N}_+$, and for each $\theta \in \Theta$, an agent checks whether $\mu_{i,t+1}(\theta) \in [0, q_{i,t}(\theta))$. If so, it encodes $\mu_{i,t+1}(\theta)$ as $q_{i,t+1}(\theta) = \mathcal{Q}_{R_{i,t}(\theta),B(\theta)}(\mu_{i,t+1}(\theta))$, with range parameter $R_{i,t}(\theta) = [0, q_{i,t}(\theta)]$, and a bit parameter $B(\theta)$ that will be specified later on. More precisely, if $\mu_{i,t+1}(\theta) \in [0, q_{i,t}(\theta))$, then $\mu_{i,t+1}(\theta)$ is encoded as:[10]

$$q_{i,t+1}(\theta) = \frac{q_{i,t}(\theta)}{2^{B(\theta)}} \lceil \mu_{i,t+1}(\theta) 2^{B(\theta)} / q_{i,t}(\theta) \rceil. \tag{6.9}$$

Agent i then broadcasts the bit-level representation of $q_{i,t+1}(\theta)$, denoted by $\bar{q}_{i,t+1}(\theta)$, to each neighbor $j \in \mathcal{N}_i$. If $\mu_{i,t+1}(\theta) \geq q_{i,t}(\theta)$, then agent i sets $q_{i,t+1}(\theta) = q_{i,t}(\theta)$, and does not broadcast about θ to any neighbor. In words, at each $t + 1 \in \mathbb{N}$, an agent

[10] Note that based on our encoding strategy, the quantized belief on any hypothesis is greater than or equal to the actual belief on that hypothesis. It is precisely this property of our quantizer that prevents beliefs on the true state from getting quantized to 0. See also Lemma 6.9.3.

Algorithm 6 (Quantized Min-Rule) Each agent $i \in \mathcal{V}$ executes this algorithm in parallel

Initialization: $\pi_{i,0}(\theta), \mu_{i,0}(\theta)$ and $\bar{\mu}_{i,0}(\theta)$ initialized as in Algorithm 5; $q_{i,0}(\theta) = 1, \forall \theta \in \Theta$.

```
for $t+1 \in \mathbb{N}_+$ do
    for $\theta \in \Theta$ do
        Update $\pi_{i,t+1}(\theta)$ via (6.2), and $\mu_{i,t+1}(\theta)$ via (6.3).
        if $\mu_{i,t+1}(\theta) \in [0, q_{i,t}(\theta))$ then
            Encode $\mu_{i,t+1}(\theta)$ via (6.9), and broadcast $\bar{q}_{i,t+1}(\theta)$ to each $j \in \mathcal{N}_i$.
        else
            Set $q_{i,t+1}(\theta) = q_{i,t}(\theta)$, and do not broadcast about $\theta$.
        end if
        for $j \in \mathcal{N}_i$ do
            if $j \in \mathcal{N}_{i,t+1}(\theta)$ then
                Decode $q_{j,t+1}(\theta)$ from $\bar{q}_{j,t+1}(\theta)$.
            else
                Set $q_{j,t+1}(\theta) = q_{j,t}(\theta)$.
            end if
        end for
        Update $\bar{\mu}_{i,t+1}(\theta)$ via (6.10).
    end for
end for
```

i broadcasts about θ if and only if $\mu_{i,t+1}(\theta)$ is strictly lower than the last quantized belief on θ that it broadcasted, namely $q_{i,t}(\theta)$. This last transmitted belief $q_{i,t}(\theta)$ also serves as the upper limit of the range $R_{i,t}(\theta)$ of the quantizer used for encoding

$\mu_{i,t+1}(\theta)$, while the lower limit remains at 0 for all time. The above steps constitute our *adaptive quantization* scheme.[11]

• **Decoding Beliefs:** For decoding beliefs, we make the following natural assumptions. For every $\theta \in \Theta$, each agent is aware of (i) the initial quantizer range, i.e., the fact that $q_{i,0}(\theta) = 1, \forall \theta \in \Theta, \forall i \in \mathcal{V}$; (ii) the nature of the encoding operation $\mathcal{Q}_{R,B}(\cdot)$; and (iii) the bit precision $B(\theta)$. Now consider any agent $j \in \mathcal{N}_i$. At any time-step $t+1 \in \mathbb{N}_+$, if j receives $\bar{q}_{i,t+1}(\theta)$ from i, then it can *exactly* recover $q_{i,t+1}(\theta)$. This follows from the assumptions we made above, and the fact that node j has access to $q_{i,t}(\theta)$, since it was the last quantized belief on θ that was transmitted by i to each of its neighbors. If j does not hear about θ from node i, then on its end, it sets $q_{i,t+1}(\theta) = q_{i,t}(\theta)$.

Based on the above discussion, it should be apparent that at each time-step $t \in \mathbb{N}$, and for each $\theta \in \Theta$, the value of $q_{i,t}(\theta)$ held by an agent i is consistent with those held by each of its neighbors - a fact that is crucial for correctly decoding the messages transmitted by i. Finally, upon completion of the decoding step, an agent i updates $\bar{\mu}_{i,t+1}(\theta)$ as:

$$\bar{\mu}_{i,t+1}(\theta) = \min\{\bar{\mu}_{i,t}(\theta), \mu_{i,t+1}(\theta), \{q_{j,t+1}(\theta)\}_{j \in \mathcal{N}_i}\}. \tag{6.10}$$

We call the above algorithm the `Quantized Min-Rule`, and outline its steps in Algorithm 6. In Line 10 of this algorithm, $\mathcal{N}_{i,t+1}(\theta)$ has the same meaning as in the rest of this paper: it represents the neighbors of i who broadcast their beliefs (in this case, quantized beliefs) on θ to i at time $t+1$.

6.6 Theoretical Guarantees for Algorithm 6

The following is our main result concerning the convergence guarantees of Algorithm 6.

[11] The adaptive nature of our encoding strategy stems from the fact that the range of the quantizer used to encode each hypothesis is dynamically updated.

Theorem 6.6.1 *Suppose every agent uses at least one bit to encode each hypothesis, i.e., let* $B(\theta) \geq 1, \forall \theta \in \Theta$. *Furthermore, suppose the following conditions hold. (i) For every pair of hypotheses* $\theta_p, \theta_q \in \Theta$, *the source set* $\mathcal{S}(\theta_p, \theta_q)$ *is non-empty. (ii) The communication graph* $\mathcal{G}$ *is connected. Then, Algorithm 6 guarantees the following.*

- ***(Consistency)****: For each agent* $i \in \mathcal{V}$, $\mu_{i,t}(\theta^\star) \to 1$ *a.s.*

- ***(Exponentially Fast Rejection of False Hypotheses)****: For each agent* $i \in \mathcal{V}$, *and for each false hypothesis* $\theta \in \Theta \setminus \{\theta^\star\}$, *the following holds:*

$$\liminf_{t\to\infty} -\frac{\log \mu_{i,t}(\theta)}{t} \geq \max_{v\in\mathcal{S}(\theta^\star,\theta)} H_v(\theta^*, \theta) \ a.s., \tag{6.11}$$

where $H_v(\theta^*, \theta) = \min\{B(\theta)\log 2, K_v(\theta^\star, \theta)\}$.

We prove the above result in Section 6.9.2. Under what conditions on $B(\theta)$ can one recover the same long-run learning rate as with infinite precision? The following result, which is an immediate corollary of Theorem 6.6.1, provides an answer.

Corollary 6.6.2 *Suppose the conditions in Theorem 6.6.1 hold. Moreover, for each* $\theta \in \Theta$, *suppose the bit precision* $B(\theta)$ *is chosen such that*

$$B(\theta) \geq \frac{1}{\log 2}\left(\max_{\theta^*\neq\theta}\max_{i\in\mathcal{V}} K_i(\theta^*, \theta)\right). \tag{6.12}$$

Then, for each $\theta \in \Theta \setminus \{\theta^\star\}$, *and* $i \in \mathcal{V}$, *we have:*

$$\liminf_{t\to\infty} -\frac{\log \mu_{i,t}(\theta)}{t} \geq \max_{v\in\mathcal{S}(\theta^\star,\theta)} K_v(\theta^\star, \theta) \ a.s. \tag{6.13}$$

We now remark on the implications of the above results.

• **1-bit precision per hypothesis is sufficient for learning:** Under standard assumptions on the observation model and the network structure, Theorem 6.6.1 reveals that based on Algorithm 6, it is possible to learn the true state exponentially fast while using just 1 bit to encode each hypothesis. Thus, at any given time-step, it suffices for each agent to broadcast an m-bit binary vector, where m is the number of hypotheses. This is a key implication of Theorem 6.6.1.

• **Trade-offs between bit-precision and learning rate:** While 1-bit precision per hypothesis is adequate for exponentially fast learning, the rate of learning may no longer be that with infinite precision. To understand this better, recall that with infinite precision, the basic min-rule in [147] allows each agent to rule out a false hypothesis θ exponentially fast at the rate $\max_{i\in\mathcal{V}} K_i(\theta^*,\theta)$.[12] Let $v \in \operatorname{argmax}_{i\in\mathcal{V}} K_i(\theta^*,\theta)$. Although agent v's belief on θ may decay to zero relatively fast, its ability to convey such a low belief to its neighbors is limited by the precision of the quantizer, when beliefs can no longer be transmitted perfectly. In particular, observe that the R.H.S. of (6.11) simplifies to $\min\{B(\theta)\log 2, \max_{i\in\mathcal{S}(\theta^*,\theta)} K_i(\theta^\star,\theta)\}$. This suggests that one can recover the same rate of rejection of θ as with infinite precision if and only if $B(\theta)\log 2 \geq \max_{i\in\mathcal{S}(\theta^*,\theta)} K_i(\theta^\star,\theta)$, i.e., a low bit-precision can come at the expense of a reduced learning rate. To sum up, just as Theorem 6.4.1 highlighted the trade-offs between sparse communication and the learning rate, Theorem 6.6.1 quantifies the trade-offs between imprecise communication and the learning rate.

• **Recovering the same learning rate as with perfect communication:** Intuitively, the condition in Eq. (6.12) can be interpreted as follows. To be able to reject $\theta \neq \theta^*$ at the same rate as with perfect communication, the range of the quantizer used to encode θ must shrink at least as fast as the fastest possible rate at which an agent can reject θ on its own, while accounting for the realization of any state $\theta^* \neq \theta$. However, in order to pick $B(\theta)$ to satisfy the condition in Eq. (6.12), an agent requires knowledge of the relative entropies of all other agents in the network. Thus, maintaining the same learning rate as with perfect communication comes at the price of global knowledge of the agents' likelihood models.

Remark 6.6.3 *Thus far, we have treated the aspects of event-triggering and quantization separately, with the aim of presenting the main algorithmic ideas and results associated with each of these themes in a clear, understandable manner. One can,*

[12] Observe that setting $B(\theta) = \infty$ in (6.11) leads to the same conclusion.

of course, combine these ideas in a variety of ways. For instance, one natural approach could be to replace the actual beliefs in the event condition of Eq. (6.4) with their quantized counterparts. Specifically, at each $t_k, k \geq 2$, an agent i checks if $\mu_{i,t_k}(\theta) < \gamma(t_k)\min\{q_{ij,t_{k-1}}(\theta), q_{ji,t_{k-1}}(\theta)\}$, where $q_{ij,t}(\theta)$ is the last quantized belief on θ transmitted by i to j up to time t.[13] *If this condition holds, i encodes $\mu_{i,t_k}(\theta)$ as:*

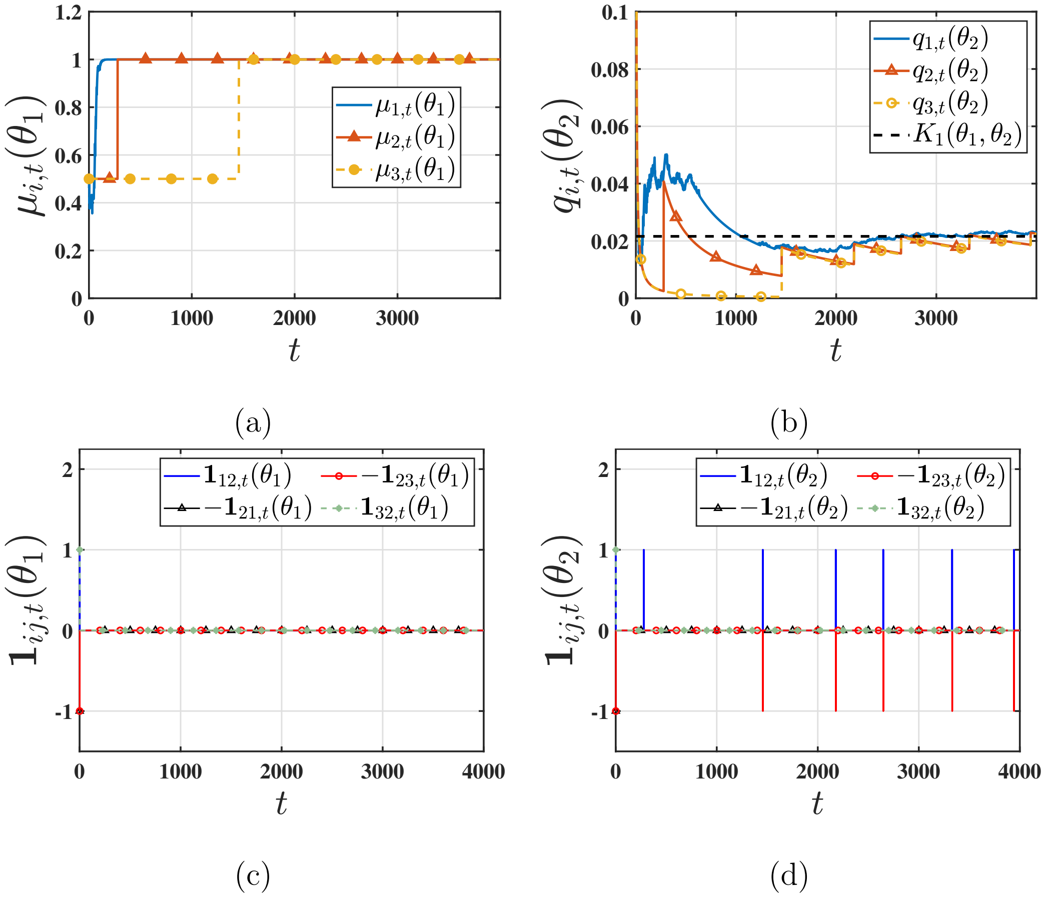
$$q_{ij,t_k}(\theta) = \frac{q_{ij,t_{k-1}}(\theta)}{2^{B(\theta)}}\lceil \mu_{i,t_k}(\theta)2^{B(\theta)}/q_{ij,t_{k-1}}(\theta)\rceil. \tag{6.14}$$

It then broadcasts $q_{ij,t_k}(\theta)$ to agent j. If the event condition does not hold, agent i sets $q_{ij,t_k}(\theta) = q_{ij,t_{k-1}}(\theta)$. It seems reasonable to expect that the above approach yields guarantees that are a blend of those in Theorems 6.4.1 and 6.6.1; we do not investigate this topic further here.

6.7 A Simulation Example

In this section, we validate our theoretical findings via a simple simulation example. To do so, we consider the network in Fig. 6.1. Suppose $\Theta = \{\theta_1, \theta_2\}, \theta^* = \theta_1$, and let the signal space for each agent be $\{0, 1\}$. The likelihood models are as follows: $l_1(0|\theta_1) = 0.7, l_1(0|\theta_2) = 0.6$, and $l_i(0|\theta_1) = l_i(0|\theta_2) = 0.5, \forall i \in \{2, 3\}$. Clearly, agent 1 is the only informative agent. To isolate the impact of our event-triggering strategy, we set $g(k) = 1, \forall k \in \mathbb{N}_+$, i.e., the event condition in Eq. 6.4 is monitored at every time-step. We set the threshold function as $\gamma(k) = 1/k^2$. The performance of Algorithm 5 is depicted in Fig. 6.2. We make the following observations. (i) From Fig. 6.2(a), we note that all agents eventually learn the truth. (ii) From Fig. 6.2(b), we note that the asymptotic rate of rejection of the false hypothesis θ_2, namely $q_{i,t}(\theta_2) = -\log(\mu_{i,t}(\theta_2))/t$, complies with the theoretical bound in Theorem 6.4.1. (iii) From Fig. 6.2(c), we note that after the first time-step, all agents stop broadcasting about the true state θ_1, complying with claim (i) of Proposition 6.4.1. (iv) From Fig. 6.2(d), we note that broadcasts about θ_2 along the path $3 \rightarrow 2 \rightarrow 1$ stop after the first

[13] Recall from Section 6.3 that the event condition (6.4) was checked at certain event-monitoring time-steps $t_k \in \mathbb{I}$.



(a)

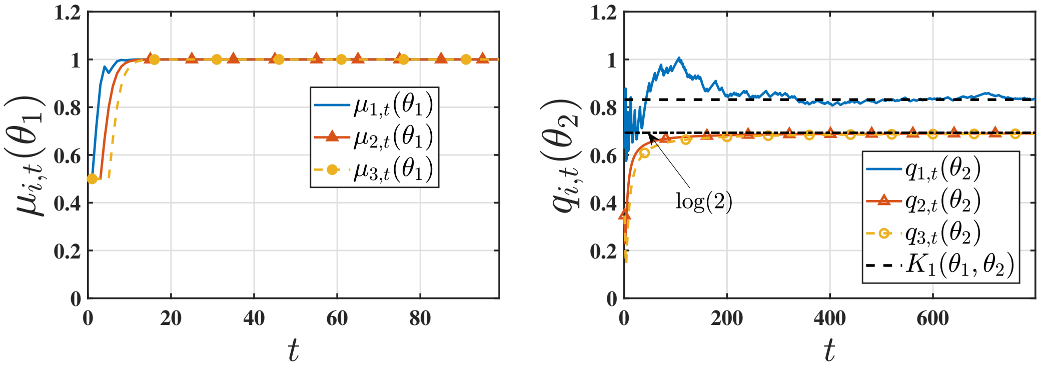
(b)

(c)

(d)

Fig. 6.2. Plots concerning the performance of Algorithm 5 for the network in Fig 6.1. Fig. 6.2(a) plots the belief evolutions on the true state θ_1. Fig. 6.2(b) plots the rate at which each agent rejects the false hypothesis θ_2, namely $q_{i,t}(\theta_2) = -\log(\mu_{i,t}(\theta_2))/t$. Fig.'s 6.2(c) and 6.2(d) demonstrate the sparse communication patterns generated by our event-triggering scheme.

time-step, in accordance with claim (ii) of Proposition 6.4.1, and Proposition 6.4.2. We also observe that in the first 4000 time-steps, agent 1 (resp., agent 2) broadcasts its belief on θ_2 to agent 2 (resp., agent 3) only 7 times (resp., 6 times). Despite such drastic reduction in the number of communication rounds, all agents still learn the truth at the best known learning rate for this problem. This demonstrates the effectiveness of our proposed framework.



(a) (b)

Fig. 6.3. Plots concerning the performance of Algorithm 6 for the network in Fig 6.1, when 1 bit is used to encode each hypothesis. Figures 6.3(a) and 6.3(b) are analogous to Figures 6.2(a) and 6.2(b). These plots demonstrate that while learning is possible even with 1-bit precision, the learning rate exhibits a dependence on the quantizer precision level.

As our second simulation study, we wish to investigate the performance of our quantized learning rule, namely Algorithm 6. To do so, keeping everything else the same, suppose we now modify the likelihood model of agent 1 as follows: $l(0|\theta_1) = 0.8$ and $l(0|\theta_2) = 0.2$. Fig. 6.3 depicts the performance of Algorithm 6 for this scenario, when $B(\theta_1) = B(\theta_2) = 1$, i.e., when 1 bit is used to encode each hypothesis. From Fig. 6.3(a), we note that all agents learn the true state. Fig. 6.3(b) reveals that the learning rates of the uninformative agents 2 and 3 are limited by the precision of the quantizer. In particular, since $K_1(\theta_1, \theta_2) = 0.8318 > \log(2)$, the learning rates for these agents get saturated at $\log(2)$, exactly as suggested by Eq. (6.11) in Theorem 6.6.1. Despite these quantization effects, we observe from Fig. 6.3(a) that the beliefs of all agents converge to the true state quite fast - a fact that is not adequately captured by our *asymptotic* learning rate analysis. This highlights the need for a finer investigation into non-asymptotic trade-offs between the learning rate and imperfect communication.

6.8 Chapter Summary

In this chapter, we developed novel communication-efficient distributed learning algorithms for addressing the distributed inference problem, subject to sparse and imprecise communication between the agents. First, we proposed an event-triggered learning rule, established that it guarantees exponentially fast convergence to the truth with probability 1, and characterized the learning rate of our rule as a function of the agents' relative entropies, the network structure, and parameters of the communication model. We discussed various trade-offs between the learning rate and the sparsity of the communication pattern, and showed that, as far as only asymptotic learning is concerned, it suffices for the agents to transmit their beliefs infinitely often, with arbitrarily long intervals between communication time-steps.

Next, we focused on scenarios where the communication channels have finite bandwidth, and developed a distributed learning rule that leverages the idea of adaptive quantization. We proved that using our rule, each agent can learn the true state exponentially fast almost surely, while using just one bit to encode its belief on each hypothesis. We established a relationship between the learning rate and the quantizer precision levels, and showed that, if the number of bits used to encode each hypothesis is chosen to be large enough, then one can in fact recover the exact same asymptotic learning rates as with infinite precision.

6.9 Omitted Proofs

6.9.1 Proofs pertaining to Section 6.4

In this section, we provide proofs of all the results stated in Section 6.4. We begin by compiling various useful properties of our update rule which will come handy during the course of our analysis.

Lemma 6.9.1 *Suppose the conditions in Theorem 6.4.1 hold. Then, there exists a set $\bar{\Omega} \subseteq \Omega$ with the following properties. (i) $\mathbb{P}^{\theta^\star}(\bar{\Omega}) = 1$. (ii) For each $\omega \in \bar{\Omega}$, there exist constants $\eta(\omega) \in (0,1)$ and $t'(\omega) \in (0,\infty)$ such that*

$$\pi_{i,t}(\theta^\star) \geq \eta(\omega), \bar{\mu}_{i,t}(\theta^\star) \geq \eta(\omega), \forall t \geq t'(\omega), \forall i \in \mathcal{V}. \tag{6.15}$$

(iii) Consider a false hypothesis $\theta \neq \theta^*$, and an agent $i \in \mathcal{S}(\theta^*, \theta)$. Then on each sample path $\omega \in \bar{\Omega}$, we have:

$$\liminf_{t\to\infty} -\frac{\log \mu_{i,t}(\theta)}{t} \geq K_i(\theta^\star, \theta). \tag{6.16}$$

Proof The proof of claim (ii) rests on the same ideas as that of [147, Lemma 2]; we thus only sketch the main arguments for completeness. From [147, Lemma 2], there exists a set $\bar{\Omega} \subseteq \Omega$ with $\mathbb{P}^{\theta^\star}(\bar{\Omega}) = 1$ such that for each $\omega \in \bar{\Omega}$, the following are true for every $i \in \mathcal{V}$: (i) $\pi_{i,t}(\theta^*) > 0, \forall t \in \mathbb{N}$; and (ii) $\exists \delta > 0$ and $t'(\omega) < \infty$ such that $\pi_{i,t}(\theta^*) \geq \delta, \forall t \geq t'(\omega)$. Fix an $\omega \in \bar{\Omega}$. Let $\rho(\omega) = \min_{i\in\mathcal{V}}\{\bar{\mu}_{i,t'(\omega)-1}(\theta^*)\}$. Based on update rules (6.1) and (6.3), observe that $\rho(\omega) > 0$; for if not, this would necessarily imply that $\pi_{i,t}(\theta^*) = 0$ for some agent i at some time-step $t \leq t'(\omega) - 1$ - a contradiction given our choice of ω. Let $\eta(\omega) = \min\{\delta, \rho(\omega)\}$, fix an agent i, and consider the update of $\mu_{i,t'(\omega)}(\theta^*)$ based on (6.3):

$$\begin{aligned} \mu_{i,t'(\omega)}(\theta^*) &= \frac{\min\{\bar{\mu}_{i,t'(\omega)-1}(\theta^*), \pi_{i,t'(\omega)}(\theta^*)\}}{\sum\limits_{p=1}^{m} \min\{\bar{\mu}_{i,t'(\omega)-1}(\theta_p), \pi_{i,t'(\omega)}(\theta_p)\}} \\ &\geq \frac{\eta(\omega)}{\sum\limits_{p=1}^{m} \pi_{i,t'(\omega)}(\theta_p)} = \eta(\omega), \end{aligned} \tag{6.17}$$

where the last equality follows from the fact that the local belief vectors generated via (6.2) are valid probability distributions over Θ at each time-step, and hence $\sum\limits_{p=1}^{m} \pi_{i,t'(\omega)}(\theta_p) = 1$. The above argument applies identically to every agent in the graph, and hence we have from (6.1) that

$$\bar{\mu}_{i,t'(\omega)}(\theta^*) = \min\{\bar{\mu}_{i,t'(\omega)-1}(\theta^*), \{\mu_{j,t'(\omega)}(\theta^*)\}_{j\in\{i\}\cup\mathcal{N}_{i,t'(\omega)}(\theta^*)}\} \geq \eta(\omega). \tag{6.18}$$

We have thus argued that for every agent $i \in \mathcal{V}$, $\mu_{i,t'(\omega)}(\theta^*) \geq \eta(\omega), \bar{\mu}_{i,t'(\omega)}(\theta^*) \geq \eta(\omega)$. We can keep repeating the above analysis for each $t > t'(\omega)$ to establish (6.15). Claim (iii) in Lemma 6.9.1 follows the same reasoning as [147, Lemma 3]. ■

The above lemma informs us that the belief $\mu_{v,t}(\theta)$ of an agent $v \in \mathcal{S}(\theta^*, \theta)$ decays exponentially fast at a rate lower-bounded by $K_v(\theta^*, \theta)$ on a set of $\mathbb{P}^{\theta^*}$-measure 1. How does this impact the belief $\mu_{i,t}(\theta)$ of an agent $i \in \mathcal{V} \setminus \mathcal{S}(\theta^*, \theta)$? The following result answers this question.

Lemma 6.9.2 *Consider a false hypothesis $\theta \in \Theta \setminus \{\theta^\star\}$ and an agent $v \in \mathcal{S}(\theta^\star, \theta)$. Suppose the conditions stated in Theorem 6.4.1 hold. Then, the following is true for each agent $i \in \mathcal{V}$:*

$$\liminf_{t\to\infty} -\frac{\log \mu_{i,t}(\theta)}{t} \geq \alpha^{d(v,i)} K_v(\theta^\star, \theta) \; a.s. \tag{6.19}$$

Proof Let $\bar{\Omega} \subseteq \Omega$ be the set of sample paths for which assertions (i)-(iii) of Lemma 6.9.1 hold. Fix a sample path $\omega \in \bar{\Omega}$, an agent $v \in \mathcal{S}(\theta^\star, \theta)$, and an agent $i \in \mathcal{V}$. When $i = v$, the assertion of Eq. (6.19) follows directly from Eq. (6.16) in Lemma 6.9.1. In particular, this implies that for a fixed $\epsilon > 0$, $\exists t_v(\omega, \theta, \epsilon)$ such that:

$$\mu_{v,t}(\theta) < e^{-(K_v(\theta^\star,\theta)-\epsilon)t}, \forall t \geq t_v(\omega, \theta, \epsilon). \tag{6.20}$$

Moreover, since $\omega \in \bar{\Omega}$, Lemma 6.9.1 guarantees the existence of a time-step $t'(\omega) < \infty$, and a constant $\eta(\omega) > 0$, such that on ω, $\pi_{i,t}(\theta^\star) \geq \eta(\omega), \bar{\mu}_{i,t}(\theta^\star) \geq \eta(\omega), \forall t \geq t'(\omega), \forall i \in \mathcal{V}$. Let $\bar{t}_v(\omega, \theta, \epsilon) = \max\{t'(\omega), t_v(\omega, \theta, \epsilon)\}$. Let $t_q > \bar{t}_v$ be the first event-monitoring time-step in $\mathbb{I}$ to the right of $\bar{t}_v$.[14] Now consider any $t_k \in \mathbb{I}$ such that $k \geq q$. In what follows, we will analyze the implications of agent v deciding whether or not to broadcast its belief on θ to a one-hop neighbor $j \in \mathcal{N}_v$ at t_k. To this end, we consider the following two cases.

[14] We will henceforth suppress the dependence of various quantities on ω, θ, and ϵ for brevity.

Case 1: $\mathbb{1}_{vj,t_k}(\theta) = 1$, i.e., v broadcasts $\mu_{v,t_k}(\theta)$ to j at t_k. Thus, since $v \in \mathcal{N}_{j,t_k}(\theta)$, we have $\bar{\mu}_{j,t_k}(\theta) \leq \mu_{v,t_k}(\theta)$ from (6.1). Let us now observe that $\forall t \geq t_k + 1$:

$$\begin{aligned} \mu_{j,t}(\theta) &\overset{(a)}{\leq} \frac{\bar{\mu}_{j,t-1}(\theta)}{\sum\limits_{p=1}^{m} \min\{\bar{\mu}_{j,t-1}(\theta_p), \pi_{j,t}(\theta_p)\}} \\ &\overset{(b)}{\leq} \frac{\mu_{v,t_k}(\theta)}{\sum\limits_{p=1}^{m} \min\{\bar{\mu}_{j,t-1}(\theta_p), \pi_{j,t}(\theta_p)\}} \overset{(c)}{<} \frac{e^{-(K_v(\theta^\star,\theta)-\epsilon)t_k}}{\eta}. \end{aligned} \tag{6.21}$$

In the above inequalities, (a) follows directly from (6.3), (b) follows by noting that the sequence $\{\bar{\mu}_{j,t}(\theta)\}$ is non-increasing based on (6.1), and (c) follows from (6.20) and the fact that all beliefs on $\theta^\star$ are bounded below by η for $t \geq \bar{t}_v$.

Case 2: $\mathbb{1}_{vj,t_k}(\theta) \neq 1$, i.e., v does not broadcast $\mu_{v,t_k}(\theta)$ to j at t_k. From the event condition in (6.4), it must then be that at least one of the following is true: (a) $\mu_{v,t_k}(\theta) \geq \gamma(t_k)\hat{\mu}_{vj,t_k}(\theta)$, and (b) $\mu_{v,t_k}(\theta) \geq \gamma(t_k)\hat{\mu}_{jv,t_k}(\theta)$. Suppose $\mu_{v,t_k}(\theta) \geq \gamma(t_k)\hat{\mu}_{vj,t_k}(\theta)$. From (6.20), we then have:

$$\hat{\mu}_{vj,t_k}(\theta) \leq \frac{\mu_{v,t_k}(\theta)}{\gamma(t_k)} < \frac{e^{-(K_v(\theta^\star,\theta)-\epsilon)t_k}}{\gamma(t_k)}. \tag{6.22}$$

In words, the above inequality places an upper bound on the belief of agent v on θ when it last transmitted its belief on θ to agent j, *prior* to time-step t_k; at least one such transmission is guaranteed to take place since all agents broadcast their entire belief vectors to their neighbors at t_1. Noting that $\bar{\mu}_{j,t}(\theta) \leq \hat{\mu}_{vj,t_k}(\theta)$, $\forall t \geq t_k$, using (6.3), (6.22), and arguments similar to those for arriving at (6.21), we obtain:

$$\mu_{j,t}(\theta) < \frac{e^{-(K_v(\theta^\star,\theta)-\epsilon)t_k}}{\eta\gamma(t_k)} \leq \frac{e^{-(K_v(\theta^\star,\theta)-\epsilon)t_k}}{\eta\gamma(t)}, \forall t \geq t_k + 1, \tag{6.23}$$

where the last inequality follows from the fact that $\gamma(\cdot)$ is a non-increasing function of its argument. Now consider the case when $\mu_{v,t_k}(\theta) \geq \gamma(t_k)\hat{\mu}_{jv,t_k}(\theta)$. Following the same reasoning as before, we can arrive at an identical upper-bound on $\hat{\mu}_{jv,t_k}(\theta)$ as in (6.22). Using the definition of $\hat{\mu}_{jv,t_k}(\theta)$, and the fact that agent j incorporates its own belief on θ in the update rule (6.1), we have that $\bar{\mu}_{j,t}(\theta) \leq \hat{\mu}_{jv,t_k}(\theta), \forall t \geq t_k$. Using similar arguments as before, observe that the bound in (6.23) holds for this case too.

Combining the analyses of cases 1 and 2, referring to (6.21) and (6.23), and noting that $\gamma(t) \in (0,1], \forall t \in \mathbb{N}$, we conclude that the bound in (6.23) holds for each $t_k \in \mathbb{I}$ such that $t_k > \bar{t}_v$. Now since $t_{k+1} - t_k = g(k)$, for any $\tau \in \mathbb{N}_+$ we have:

$$t_{q+\tau} = t_q + \sum_{z=q}^{q+\tau-1} g(z). \tag{6.24}$$

Next, noting that $g(\cdot)$ is non-decreasing, observe that:

$$t_q + \int_q^{q+\tau} g(z-1)dz \leq t_{q+\tau} \leq t_q + \int_q^{q+\tau} g(z)dz. \tag{6.25}$$

The above yields: $l(q,\tau) \triangleq t_q + G(q+\tau-1) - G(q-1) \leq t_{q+\tau} \leq t_q + G(q+\tau) - G(q) \triangleq u(q,\tau)$. Fix any time-step $t > u(q,1)$, let $\tau(t)$ be the largest index such that $u(q,\tau(t)) < t$, and $\bar{\tau}(t)$ be the largest index such that $t_{q+\bar{\tau}(t)} < t$. Observe:

$$\bar{t}_v < t_q < t_{q+1} \leq t_{q+\tau(t)} \leq t_{q+\bar{\tau}(t)} < t. \tag{6.26}$$

Using the above inequality, the fact that $l(q,\tau(t)) \leq t_{q+\tau(t)}$, and referring to (6.23), we obtain:

$$\mu_{j,t}(\theta) < \frac{e^{-(K_v(\theta^\star,\theta)-\epsilon)t_{q+\bar{\tau}(t)}}}{\eta\gamma(t)} \leq \frac{e^{-(K_v(\theta^\star,\theta)-\epsilon)l(q,\tau(t))}}{\eta\gamma(t)} \tag{6.27}$$

From the definition of $\tau(t)$, we have $q + \tau(t) = \lceil G^{-1}(t - t_q + G(q)) \rceil - 1$. This yields:

$$\begin{aligned} l(q,\tau(t)) &= t_q + G(\lceil G^{-1}(t - t_q + G(q)) \rceil - 2) - G(q-1) \\ &\geq t_q + G(G^{-1}(t - t_q + G(q)) - 2) - G(q-1). \end{aligned} \tag{6.28}$$

From (6.27) and (6.28), we obtain the following $\forall t > u(q,1)$:

$$-\frac{\log \mu_{j,t}(\theta)}{t} > \frac{\tilde{G}(t)}{t}(K_v(\theta^\star,\theta) - \epsilon) - \frac{\log c}{t} - \frac{\log(1/\gamma(t))}{t}, \tag{6.29}$$

where $\tilde{G}(t) = G(G^{-1}(t - t_q + G(q)) - 2)$, and $c = e^{-(K_v(\theta^\star,\theta)-\epsilon)(t_q - G(q-1))}/\eta$. Now taking the limit inferior on both sides of (6.29) and using (6.5) yields:

$$\liminf_{t\to\infty} -\frac{\log \mu_{j,t}(\theta)}{t} \geq \alpha(K_v(\theta^\star,\theta) - \epsilon). \tag{6.30}$$

Finally, since the above inequality holds for any sample path $\omega \in \bar{\Omega}$, and an arbitrarily small ϵ, it follows that the assertion in (6.19) is true for every one-hop neighbor j of agent v.

Now consider any agent i such that $d(v,i) = 2$. Clearly, there must exist some $j \in \mathcal{N}_v$ such that $i \in \mathcal{N}_j$. Following an identical line of reasoning as before, it is easy to see that with $\mathbb{P}^{\theta^*}$-measure 1, $\mu_{i,t}(\theta)$ decays exponentially at a rate that is at least α times the rate at which $\mu_{j,t}(\theta)$ decays to zero. From (6.30), the latter rate is at least $\alpha K_v(\theta^*, \theta)$, and hence, the former is at least $\alpha^2 K_v(\theta^*, \theta)$. This establishes the claim of the lemma for all agents that are two-hops away from agent v. Since $\mathcal{G}$ is connected, given any $i \in \mathcal{V}$, there exists a path $\mathcal{P}(v,i)$ in $\mathcal{G}$ from v to i. One can keep repeating the above argument along the path $\mathcal{P}(v,i)$ and proceed via induction to complete the proof. ■

We are now in position to prove Theorem 6.4.1.

Proof (**Theorem 6.4.1**) Fix a $\theta \in \Theta \setminus \{\theta^\star\}$. Based on condition (i) of the Theorem, $\mathcal{S}(\theta^\star, \theta)$ is non-empty, and based on condition (ii), there exists a path from each agent $v \in \mathcal{S}(\theta^\star, \theta)$ to every agent $i \in \mathcal{V} \setminus \{v\}$; Eq. (6.6) then follows from Lemma 6.9.2. By definition of a source set, $K_v(\theta^\star, \theta) > 0, \forall v \in \mathcal{S}(\theta^\star, \theta)$; Eq. (6.6) then implies $\lim_{t\to\infty} \mu_{i,t}(\theta) = 0$ a.s., $\forall i \in \mathcal{V}$. ■

Proof (**Proposition 6.4.1**) Let the set $\bar{\Omega}$ have the same meaning as in Lemma 6.9.2. Fix any $\omega \in \bar{\Omega}$, and note that since the conditions of Theorem 6.4.1 are met, $\mu_{i,t}(\theta^*) \to 1$ on $\omega, \forall i \in \mathcal{V}$. We prove the first claim of the proposition via contradiction. Accordingly, suppose the claim does not hold. Since there are only finitely many agents, this implies the existence of some $i \in \mathcal{V}$ and some $j \in \mathcal{N}_i$, such that i broadcasts its belief on θ^* to j infinitely often, i.e., there exists a sub-sequence $\{t_{p_k}\}$ of $\{t_k\}$ at which the event-condition (6.4) gets satisfied for θ^*. From (6.4), $\mu_{i,t_{p_k}}(\theta^*) < \gamma^k \mu_{i,t_{p_0}}(\theta^*), \forall k \in \mathbb{N}_+$, where $\gamma \triangleq \gamma(t_{p_0})$. This implies $\lim_{k\to\infty} \mu_{i,t_{p_k}}(\theta^*) = 0$, contradicting the fact that on ω, $\lim_{t\to\infty} \mu_{i,t}(\theta^*) = 1$.

For establishing the second claim, fix $\omega \in \bar{\Omega}$, $\theta \neq \theta^*$, and $i \notin \mathcal{S}(\theta^*, \theta)$. Since $i \notin \mathcal{S}(\theta^*, \theta)$, there exists $\tilde{t}_1 < \infty$ and $\bar{\eta} > 0$, such that $\pi_{i,t}(\theta) \geq \bar{\eta}, \forall t \geq \tilde{t}_1$. This follows from the fact that since θ is observationally equivalent to θ^* for agent i, the claim regarding $\pi_{i,t}(\theta^*)$ in Eq. (6.15) applies identically to $\pi_{i,t}(\theta)$. Note also that since the conditions of Theorem 6.4.1 are met, $\mu_{i,t}(\theta) \to 0$ on ω. From (6.1), $\bar{\mu}_{i,t}(\theta) \to 0$ as well. Thus, there must exist some $\tilde{t}_2 < \infty$ such that $\min\{\bar{\mu}_{i,t}(\theta), \pi_{i,t+1}(\theta)\} = \bar{\mu}_{i,t}(\theta), \forall t \geq \tilde{t}_2$. Let $\tilde{t} = \max\{\tilde{t}_1, \tilde{t}_2\}$. Consider any $t_k \in \mathbb{I}, t_k > \tilde{t}$. We claim:

$$\mu_{i,t}(\theta) \geq \bar{\mu}_{i,t_k}(\theta), \forall t \in [t_k + 1, t_{k+1}], \text{ and} \tag{6.31}$$

$$\bar{\mu}_{i,t}(\theta) \geq \bar{\mu}_{i,t_k}(\theta), \forall t \in [t_k, t_{k+1}). \tag{6.32}$$

To see why the above inequalities hold, consider the update of $\mu_{i,t_k+1}(\theta)$ based on (6.3). Since $t_k > \tilde{t}_2$, we have $\min\{\bar{\mu}_{i,t_k}(\theta), \pi_{i,t_k+1}(\theta)\} = \bar{\mu}_{i,t_k}(\theta)$. Noting that the denominator of the fraction on the R.H.S. of (6.3) is at most 1, we obtain: $\mu_{i,t_k+1}(\theta) \geq \bar{\mu}_{i,t_k}(\theta)$. If $t_k+1 = t_{k+1}$, then the claim follows. Else, if $t_k+1 < t_{k+1}$, then since no communication occurs at t_k+1, we have from (6.1) that $\bar{\mu}_{i,t_k+1}(\theta) = \min\{\bar{\mu}_{i,t_k}(\theta), \mu_{i,t_k+1}(\theta)\} \geq \bar{\mu}_{i,t_k}(\theta)$. We can keep repeating the above argument for each $t \in [t_k + 1, t_{k+1}]$ to establish the claim. In words, inequalities (6.31) and (6.32) reveal that agent i cannot lower its belief on the false hypothesis θ between two consecutive event-monitoring time-steps when it does not hear from any neighbor. We will make use of this fact repeatedly during the remainder of the proof. Let $t_{p_0} > \tilde{t}$ be the first time-step in $\mathbb{I}$ to the right of $\tilde{t}$. Now consider the following sequence, where $k \in \mathbb{N}$:

$$t_{p_{k+1}} = \inf\{t \in \mathbb{I} : t > t_{p_k}, \bar{\mu}_{i,t}(\theta) < \bar{\mu}_{i,t-1}(\theta)\}. \tag{6.33}$$

The above sequence represents those event-monitoring time-steps at which $\bar{\mu}_{i,t}(\theta)$ decreases. We first argue that $\{t_{p_k}\}$ is well-defined, i.e., each term in the sequence is finite. If not, then based on (6.32), this would mean that $\bar{\mu}_{i,t}(\theta)$ remains bounded away from 0, contradicting the fact that $\bar{\mu}_{i,t}(\theta) \to 0$ on ω. Next, for each $k \in \mathbb{N}_+$, let $j_{p_k} \in \operatorname{argmin}_{j \in \mathcal{N}_{i,t_{p_k}}(\theta) \cup \{i\}} \mu_{j,t_{p_k}}(\theta)$. We claim that $i \neq j_{p_k}$. To see why this is true, suppose, if possible, $i = j_{p_k}$. Then, based on the definition of t_{p_k}, we would have

$\bar{\mu}_{i,t_{p_k}}(\theta) = \mu_{i,t_{p_k}}(\theta) < \bar{\mu}_{i,t_{p_k}-1}(\theta)$. However, as $t_{p_k} > \tilde{t}_2$, we have from (6.3) that $\mu_{i,t_{p_k}}(\theta) \geq \bar{\mu}_{i,t_{p_k}-1}(\theta)$, leading to the desired contradiction. In the final step of the proof, we claim that i does not broadcast its belief on θ to j_{p_k} over $[t_{p_k}+1, t_{p_{k+1}}]$.

To establish this claim, we start by noting that based on the definitions of j_{p_k} and t_{p_k}, $\bar{\mu}_{i,t_{p_k}}(\theta) = \mu_{j_{p_k},t_{p_k}}(\theta)$. Let us first consider the case when there are no intermediate event-monitoring time-steps in $(t_{p_k}, t_{p_{k+1}})$, i.e., t_{p_k} and $t_{p_{k+1}}$ are consecutive terms in $\mathbb{I}$. Then, at $t_{p_{k+1}}$, $\hat{\mu}_{j_{p_k}i,t_{p_{k+1}}}(\theta) = \mu_{j_{p_k},t_{p_k}}(\theta)$, since no communication occurs over $(t_{p_k}, t_{p_{k+1}})$. Moreover, using (6.31), $\mu_{i,t_{p_{k+1}}}(\theta) \geq \bar{\mu}_{i,t_{p_k}}(\theta) = \mu_{j_{p_k},t_{p_k}}(\theta)$. Thus, the event condition (6.4) gets violated at $t_{p_{k+1}}$, and i does not broadcast its belief on θ to j_{p_k}. Next, consider the scenario when there is exactly one event-monitoring time-step - say $\bar{t} \in \mathbb{I}$ - in the interval $(t_{p_k}, t_{p_{k+1}})$. Since t_{p_k} and $\bar{t}$ are now consecutive terms in $\mathbb{I}$, the fact that $\mathbb{1}_{ij_{p_k},\bar{t}}(\theta) \neq 1$ follows from exactly the same reasoning as earlier. We argue that $\mathbb{1}_{j_{p_k}i,\bar{t}}(\theta) \neq 1$ as well. To see this, suppose that j_{p_k} does in fact broadcast $\mu_{j_{p_k},\bar{t}}(\theta)$ to i at $\bar{t}$. For this to happen, the event condition (6.4) entails: $\mu_{j_{p_k},\bar{t}}(\theta) < \gamma(\bar{t})\mu_{j_{p_k},t_{p_k}}(\theta) = \gamma(\bar{t})\bar{\mu}_{i,t_{p_k}}(\theta) \leq \bar{\mu}_{i,t_{p_k}}(\theta)$. Since $\bar{\mu}_{i,\bar{t}-1}(\theta) \geq \bar{\mu}_{i,t_{p_k}}(\theta)$ from (6.32), $\mathbb{1}_{j_{p_k}i,\bar{t}}(\theta) = 1$ would then imply that $\bar{\mu}_{i,\bar{t}}(\theta) < \bar{\mu}_{i,\bar{t}-1}(\theta)$, violating the fact that $\bar{t} < t_{p_{k+1}}$. The above reasoning suggests that $\hat{\mu}_{j_{p_k}i,t}(\theta) = \mu_{j_{p_k},t_{p_k}}(\theta), \forall t \in (t_{p_k}, t_{p_{k+1}}]$. Moreover, since $\bar{\mu}_{i,t}(\theta)$ does not decrease at $\bar{t}$ (as $\bar{t} < t_{p_{k+1}}$), we have from (6.31) that $\mu_{i,t}(\theta) \geq \bar{\mu}_{i,t_{p_k}}(\theta) = \mu_{j_{p_k},t_{p_k}}(\theta), \forall t \in (t_{p_k}, t_{p_{k+1}}]$. It follows from the preceding discussion that (6.4) gets violated at $t_{p_{k+1}}$, and hence $\mathbb{1}_{ij_{p_k},t_{p_{k+1}}}(\theta) \neq 1$. The above arguments readily carry over to the case when there are an arbitrary number of event-monitoring time-steps in the interval $(t_{p_k}, t_{p_{k+1}})$. Thus, we omit such details.

We conclude that over each interval of the form $(t_{p_k}, t_{p_{k+1}}], k \in \mathbb{N}_+$, there exists a neighbor $j_{p_k} \in \mathcal{N}_i$ to which agent i does not broadcast its belief on θ. We can obtain one such t_{p_1} for each $i \notin \mathcal{S}(\theta^*, \theta)$, and take the maximum of such time-steps to obtain $T_2(\omega)$. ■

Proof (**Proposition 6.4.2**) Let us fix $\theta \neq \theta^*$, and partition the set of agents $\mathcal{V} \setminus \{v_\theta\}$ based on their distances from v_θ. Accordingly, we use $\mathcal{L}_q(\theta)$ to represent level-q agents that are at distance q from v_θ, where $q \in \mathbb{N}_+$. Let the agent(s) that are farthest from

v_θ be at level $\bar{q}$. Now consider any agent $i \in \mathcal{L}_{\bar{q}}(\theta)$. Based on the conditions of the proposition, note that $i \notin \mathcal{S}(\theta^*, \theta)$, and the only neighbor of i is its parent in the tree rooted at v_θ, denoted by $p_i(\theta)$. Thus, claim (ii) of Proposition 6.4.1 applies to agent i, implying that agent i stops broadcasting its belief on θ to $p_i(\theta)$ eventually almost surely. Next, consider an agent $j \in \mathcal{L}_{\bar{q}-1}(\theta)$. We have already argued that after a finite number of time-steps, j will stop hearing broadcasts about θ from its children in level $\bar{q}$. Thus, for large enough k, $\mathcal{N}_{j,t_k}(\theta)$ can only comprise of $p_j(\theta)$, namely the parent of agent j in level $\bar{q}-2$. In particular, given that $j \notin \mathcal{S}(\theta^*, \theta)$, the decrease in $\bar{\mu}_{j,t}(\theta)$ at time-steps defined by (6.33) can only be caused by $p_j(\theta)$. It then readily follows from the proof of Proposition 6.4.1 that j will stop broadcasting $\mu_{j,t}(\theta)$ to $p_j(\theta)$ eventually almost surely. To complete the proof, we can keep repeating the above argument till we reach level 1. ■

Proof (**Theorem** 6.4.3) The proof of this result is similar in spirit to that of Theorem 6.4.1. Hence, we only sketch the essential details. We begin by noting that the claims in Lemma 6.9.1 hold under the conditions of the theorem - this can be easily verified. Let $\bar{\Omega}$ have the same meaning as in Lemma 6.9.2. Fix $\omega \in \bar{\Omega}$ and an arbitrarily small $\epsilon > 0$. Since $\mathbb{P}^{\theta^*}(\bar{\Omega}) = 1$, to prove the result, it suffices to argue that for each false hypothesis $\theta \neq \theta^*$, $\exists T(\omega, \theta, \epsilon)$ such that on ω, $\mu_{i,t}(\theta) < \epsilon, \forall t \geq T(\omega, \theta, \epsilon), \forall i \in \mathcal{V}$. Recall that based on Lemma 6.9.1, there exists a time-step $t'(\omega) < \infty$, and a constant $\eta(\omega) > 0$, such that on ω, $\pi_{i,t}(\theta^\star) \geq \eta(\omega), \bar{\mu}_{i,t}(\theta^\star) \geq \eta(\omega), \forall t \geq t'(\omega), \forall i \in \mathcal{V}$. Set $\bar{\epsilon}(\omega) = \min\{\epsilon, \gamma\eta(\omega)\}$. Also, from Lemma 6.9.1, we know that there exists $\bar{t}$ such that $\mu_{i,t}(\theta) < \bar{\epsilon}^{|\mathcal{V}|}, \forall t \geq \bar{t}, \forall i \in \mathcal{S}(\theta^*, \theta)$.[15] Let $\tilde{t}_0 = \max\{t', \bar{t}\}$. Since the union graph over $[\tilde{t}_0, \infty)$ is rooted at $\mathcal{S}(\theta^*, \theta)$, there exists a set $\mathcal{F}_1(\theta) \in \mathcal{V} \setminus \mathcal{S}(\theta^*, \theta)$ of agents such that each agent in $\mathcal{F}_1(\theta)$ has at least one neighbor in $\mathcal{S}(\theta^*, \theta)$ in the union graph. Accordingly, consider any $j \in \mathcal{F}_1(\theta)$, and suppose $j \in \mathcal{N}_i(\tau)$, for some $i \in \mathcal{S}(\theta^*, \theta)$,

[15] As before, we have suppressed dependence of various quantities on ω, θ, and ϵ, since they can be inferred from context.

and some $\tau \geq \tilde{t}_0$. The cases $\mathbb{1}_{ij,\tau}(\theta) = 1$ and $\mathbb{1}_{ij,\tau}(\theta) \neq 1$ can be analyzed exactly as in the proof of Lemma 6.9.2 to yield:

$$\mu_{j,t}(\theta) < \frac{\bar{\epsilon}^{|\mathcal{V}|}}{\eta\gamma} \leq \bar{\epsilon}^{(|\mathcal{V}|-1)}, \forall t > \tau, \tag{6.34}$$

where the last inequality follows by noting that $\bar{\epsilon} \leq \eta\gamma$. Let $\tilde{t}_1 > \tilde{t}_0$ be the first time-step by which every agent in $\mathcal{F}_1(\theta)$ has had at least one neighbor in $\mathcal{S}(\theta^*, \theta)$. Then, based on the above reasoning, $\mu_{j,t}(\theta) < \bar{\epsilon}^{(|\mathcal{V}|-1)}, \forall t > \tilde{t}_1, \forall j \in \mathcal{F}_1(\theta)$. If $\mathcal{V} \setminus \{\mathcal{S}(\theta^*, \theta) \cup \mathcal{F}_1(\theta)\} = \emptyset$, then we are done. Else, given the fact that the union graph over $[\tilde{t}_1, \infty)$ is rooted at $\mathcal{S}(\theta^*, \theta)$, there must exist a non-empty set $\mathcal{F}_2(\theta)$ such that each agent in $\mathcal{F}_2(\theta)$ has at least one neighbor from the set $\mathcal{S}(\theta^*, \theta) \cup \mathcal{F}_1(\theta)$ in the union graph. Reasoning as before, one can conclude that there exists a time-step $\tilde{t}_2 > \tilde{t}_1$ such that $\mu_{j,t}(\theta) < \bar{\epsilon}^{(|\mathcal{V}|-2)}, \forall t > \tilde{t}_2, \forall j \in \mathcal{F}_2(\theta)$. To complete the proof, we can keep repeating the above construction till we exhaust the vertex set $\mathcal{V}$. ■

6.9.2 Proof of Theorem 6.6.1

We begin with the following lemma.

Lemma 6.9.3 *Suppose the conditions of Theorem 6.6.1 are satisfied. Then, assertions (i)-(iii) in Lemma 6.9.1 hold when each agent employs Algorithm 6.*

Proof The proof of this lemma mirrors that of Lemma 6.9.1. The key point is that for any agent $i \in \mathcal{V}$, $q_{i,t}(\theta^*) \neq 0$ almost surely, where $t \in \mathbb{N}$. To see this, observe from (6.9) that whenever an agent i broadcasts about θ^*, we have $q_{i,t}(\theta^*) \geq \mu_{i,t}(\theta^*)$. Hence, at such a time-step t, $q_{i,t}(\theta^*) = 0 \implies \mu_{i,t}(\theta^*) = 0$. Using the same arguments as in Lemma 6.9.1, one can argue that this is almost surely impossible. ■

Equipped with the above lemma, we now proceed to prove Theorem 6.6.1.

Proof (**Theorem 6.6.1**) In view of Lemma 6.9.3, we know that there exists a set $\bar{\Omega} \subseteq \Omega$ of $\mathbb{P}^{\theta^*}$-measure 1 for which assertions (ii) and (iii) of Lemma 6.9.1 hold. Consider any false hypothesis $\theta \neq \theta^*$, fix a sample path $\omega \in \bar{\Omega}$, and an agent $v \in$

$\mathcal{S}(\theta^\star, \theta)$. Following the same reasoning as in the proof of Lemma 6.9.2, there exists a time-step $\bar{t}$, such that for all $t \geq \bar{t}$, the following are true on ω: (i) $\pi_{i,t}(\theta^\star) \geq \eta(\omega), \bar{\mu}_{i,t}(\theta^\star) \geq \eta(\omega), \forall i \in \mathcal{V}$; and (ii) for a fixed $\epsilon > 0$, $\mu_{v,t}(\theta) < e^{-(K_v(\theta^\star,\theta)-\epsilon)t}$. We will complete the proof in two steps. In Step 1, we will establish that the quantization range $R_{v,t}(\theta) = [0, q_{v,t}(\theta)]$ contracts exponentially fast. In Step 2, we will analyze the implications of the above phenomenon on the beliefs of the remaining agents on θ. In what follows, we elaborate on these steps.

Step 1. Consider any time-step $t + 1 > \bar{t}$. At this time-step, there are two possibilities. Either $\mu_{v,t+1}(\theta) \in [0, q_{v,t}(\theta))$, in which case we have from (6.9) that:

$$q_{v,t+1}(\theta) \leq \frac{1}{2^{B(\theta)}} q_{v,t}(\theta) + \mu_{v,t+1}(\theta). \tag{6.35}$$

Else, we have $\mu_{v,t+1}(\theta) \geq q_{v,t}(\theta)$ and, based on our encoding strategy, node v sets $q_{v,t+1}(\theta) = q_{v,t}(\theta)$. Clearly, the bound on $q_{v,t+1}(\theta)$ in (6.35) applies to both the cases we discussed above. To proceed, let $a = 1/2^{B(\theta)}$, $\tilde{K} = K_v(\theta^*, \theta) - \epsilon$, and $\rho = \max\{a, e^{-\tilde{K}}\}$. Rolling out the inequality (6.35) over $\tau \geq 1$ time-steps starting from $\bar{t}$ yields:

$$\begin{aligned} q_{v,\bar{t}+\tau}(\theta) &\leq a^\tau \left(q_{v,\bar{t}}(\theta) + \sum_{l=0}^{\tau-1} \frac{\mu_{v,\bar{t}+l+1}}{a^{l+1}} \right) \\ &\overset{(a)}{\leq} a^\tau \left(q_{v,\bar{t}}(\theta) + \frac{e^{-\tilde{K}(\bar{t}+1)}}{a} \sum_{l=0}^{\tau-1} \frac{1}{\left(ae^{\tilde{K}}\right)^l} \right) \\ &\overset{(b)}{\leq} a^\tau + \frac{e^{-\tilde{K}\tau} - a^\tau}{e^{-\tilde{K}} - a} \\ &\overset{(c)}{\leq} \left(1 + \frac{1}{|e^{-\tilde{K}} - a|} \right) \rho^\tau. \end{aligned} \tag{6.36}$$

In the above inequalities, (a) follows by noting that $\mu_{v,\bar{t}+l+1}$ decays exponentially $\forall l \geq 0$ based on the definition of $\bar{t}$. For (b), we simplify the preceding inequality using the facts that $q_{v,\bar{t}}(\theta) \leq 1$, and $e^{-\tilde{K}(\bar{t}+1)} \leq 1$ as $\tilde{K} > 0$; the latter is true since $v \in \mathcal{S}(\theta^*, \theta)$. Finally, (c) follows from straightforward algebra. We thus obtain:

$$q_{v,t}(\theta) \leq \frac{1}{\rho^{\bar{t}}} \left(1 + \frac{1}{|e^{-\tilde{K}} - a|} \right) \rho^t, \forall t \geq \bar{t} + 1. \tag{6.37}$$

Since $B(\theta) \geq 1$, we have $a < 1$. Moreover, as $\tilde{K} > 0$, it follows that $\rho < 1$. In view of (6.37), we thus observe that $q_{v,t}(\theta)$ eventually decays to 0 exponentially fast at the rate ρ.

Step 2. Consider any neighbor j of agent v. Let us now make two simple observations, each of which follow easily from the rules of Algorithm 6. First, given that $\mu_{v,1}(\theta) < 1 = q_{v,0}(\theta)$, the condition in line 4 of Algorithm 6 will pass at $t = 1$, and hence agent v will broadcast $q_{v,1}(\theta)$ to agent j at time-step $t = 1$. Second, at each following time-step $t \geq 1$, the value of $q_{v,t}(\theta)$ held by agent v is consistent with that held by agent j, irrespective of whether v broadcasts to j at time t about θ, or not. We thus have that $\forall t \geq \bar{t} + 2$:

$$\begin{aligned}\mu_{j,t}(\theta) &\overset{(a)}{\leq} \frac{\bar{\mu}_{j,t-1}(\theta)}{\eta} \\ &\overset{(b)}{\leq} \frac{q_{v,t-1}(\theta)}{\eta} \\ &\overset{(c)}{\leq} \frac{1}{\eta \rho^{\bar{t}+1}} \left(1 + \frac{1}{|e^{-\tilde{K}} - a|}\right) \rho^{t},\end{aligned} \tag{6.38}$$

where (a) follows from (6.3) and the fact that all beliefs on θ^* are bounded below by η for $t \geq \bar{t}$; (b) follows from (6.10); and (c) follows from (6.37). Taking the natural log on both sides of (6.38), dividing throughout by t, and then taking the limit inferior on both sides of the resulting inequality yields:

$$\liminf_{t \to \infty} -\frac{\log \mu_{j,t}(\theta)}{t} \geq \log \frac{1}{\rho}. \tag{6.39}$$

Now let us consider two cases. First, suppose $B(\theta) \log 2 \geq K_v(\theta^*, \theta)$. From (6.39), it is then easy to see that $\log 1/\rho \geq \tilde{K} = K_v(\theta^*, \theta) - \epsilon$, where ϵ can be made arbitrarily small. Hence, in this case, $\log 1/\rho \geq K_v(\theta^*, \theta)$. Next, suppose $B(\theta) \log 2 < K_v(\theta^*, \theta)$. Then, there must exist $\epsilon > 0$ such that $B(\theta) \log 2 < K_v(\theta^*, \theta) - \epsilon$. With such a choice of ϵ, we can set $\tilde{K} = K_v(\theta^*, \theta) - \epsilon$ and conduct the above analysis to arrive at $\log 1/\rho \geq B(\theta) \log 2$. Hence, we conclude:

$$\liminf_{t \to \infty} -\frac{\log \mu_{j,t}(\theta)}{t} \geq \min\{B(\theta) \log 2, K_v(\theta^*, \theta)\}. \tag{6.40}$$

Consider any neighbor l of agent j, i.e., a two-hop neighbor of agent v. We can analyze the decay of $q_{j,t}(\theta)$ and $\mu_{l,t}(\theta)$ exactly as we did for $q_{v,t}(\theta)$ and $\mu_{j,t}(\theta)$ to conclude that $\mu_{l,t}(\theta)$ also decays exponentially at a rate that is lower bounded by $H_v(\theta^*, \theta) = \min\{B(\theta)\log 2, K_v(\theta^*, \theta)\}$; this is not too hard to verify and hence we omit details. Repeating this argument reveals that every agent reachable from v can reject θ at a rate that is at least $H_v(\theta^*, \theta)$. Since $\mathcal{G}$ is connected, the above conclusion applies to every agent.

An analysis identical to the one above can be carried out for each $v \in \mathcal{S}(\theta^*, \theta)$. The proof can then be completed following the same arguments as in Theorem 6.4.1.

■

7. SUMMARY AND FUTURE DIRECTIONS

In this thesis, we focused on understanding how to solve estimation and inference problems over networks with distributed data. In the first part of the thesis, we made the following contributions to the problem of distributed state estimation.

- In Chapter 2, we introduced a new class of distributed observers that guarantee asymptotic reconstruction of the state under minimal system- and graph-theoretic assumptions. We also studied the problem of designing distributed functional observers.

- In Chapter 3, we identified (separate) necessary and sufficient conditions for addressing the distributed state estimation problem subject to worst-case Byzantine adversarial attacks on certain agents.

- In Chapter 4, we extended our framework to accommodate a broad class of time-varying graphs. In particular, we proved that our approach guarantees exponential convergence at any desired rate, under remarkably mild assumptions on the sequence of time-varying graphs.

In the second part of the thesis, we studied the problem of statistical inference over a network, and contributed in the following ways.

- In Chapter 5, we introduced a new distributed learning rule based on a min-protocol. We established that it guarantees exponentially fast convergence with probability 1 at a network-independent rate that strictly improves upon the rates existing in the literature. We then developed a simple and efficient variant of this rule that can account for the presence of misbehaving entities.

- Finally, in Chapter 6, we explored the theme of communication-efficiency. To reduce the number of communication rounds, we introduced an event-triggered learning algorithm, and characterized the trade-offs between sparse communication and the learning rate. To tackle the aspect of finite channel bandwidth, we drew on ideas from adaptive quantization to develop a quantized learning rule; we showed that this rule guarantees exponential convergence almost surely even when just 1 bit is used to encode each hypothesis.

Let us now briefly outline certain problems that are a subject of both ongoing and future research.

- **Distributed Observer Design beyond LTI dynamics**: In our developments pertaining to distributed state estimation, the standing assumption has been that the state dynamics evolves based on an LTI model. Working under this standing assumption on the dynamical model, we were able to essentially establish that a distributed observer can match almost all the properties of its centralized counterpart. We then generalized our results to account for time-varying graphs and worst-case adversarial attacks on the network. While these developments provide a fairly complete theory for distributed state estimation of LTI systems, generalizations along the dynamical system model remain open. In particular, do analogous ideas and results carry over to certain classes of nonlinear systems? What about switched linear systems? While there are some recent results [179,180] on the design of distributed observers for certain classes of nonlinear systems, there remains much to explore along these lines.

  Coming back to the realm of LTI systems, one can also study dynamics of the form $\mathbf{x}[k+1] = \mathbf{A}\mathbf{x}[k] + \mathbf{B}\mathbf{u}[k]$, where $\mathbf{u}[k]$ is an unknown input. Suppose the pair $(\mathbf{A}, \mathbf{C})$ is strongly-detectable, where $\mathbf{C}$ is the collective observation matrix. Furthermore, suppose the communication graph $\mathcal{G}$ is strongly-connected. Under these assumptions, is it true that one can construct a distributed unknown input and state observer? To the best of our knowledge, this question remains completely unexplored.

In addition to the above problems, one can also study the effect of a time-varying measurement model at each node. Such an abstraction is particularly relevant in the context of environmental monitoring using mobile sensors. Since a mobile sensor persistently moves around between sensing locations, its measurement model becomes a function of its motion trajectory. As a result, even if the underlying state dynamics evolves based on an LTI model, the measurement model for each sensor is time-varying. Under such circumstances, how does one design a patrol that allows each sensor to estimate the state despite the issue of intermittent observations? This is essentially a switched system observer design problem. While we have conducted preliminary investigations along this line in [117, 181], there are various questions that remain unanswered.

- **Closing the Loop:** The investigations in this thesis have been directed towards the problem of estimating/inferring an unknown quantity with distributed data. A natural follow-up problem would be to use the estimate of the state to control some process of interest. For instance, suppose the task is to *jointly* estimate and control the state of an LTI system. Can this be done if the state is jointly detectable and stabilizable w.r.t. the measurements and inputs of all the nodes? Note that there may be circumstances where the actuator nodes in the network are separated from the sensing nodes that acquire measurements. Given this predicament, can one identify the exact graph conditions that lead to a solution to this problem? A preliminary result on this topic can be found here [182].

- **Sequential Decision-Making in Multi-Agent Systems:** As a generalization of the above point, one can ask: How should an agent *act* based on the information it has acquired over time? Depending upon the context, various formulations can be conceived. For instance, suppose an individual wishes to determine the best restaurant in a city. Each visit to a restaurant reveals a noisy estimate of the restaurant's quality. In the language of online learning, the goal of the individual is to minimize regret by carefully trading off between exploration and exploitation. This is of course a very well studied problem in the multi-armed bandits literature. Now suppose the individual in the previous example can benefit from side observations made by her friends in the city. Can she leverage such additional information to achieve a lower regret than when she had to act alone? If so, what should be her strategy of data-aggregation? If each query made to a friend incurs a cost, how would her strategy change? More generally, how should a network of agents interact to resolve the exploration-exploitation dilemma inherent in online learning problems of the form stated above? Some recent results on this topic can be found in [183–185].

- **Communication-Efficient Distributed Learning and Control:** Given a task that needs to be accomplished in a distributed manner (e.g., control, estimation, optimization, or learning), how often should an agent interact with its neighbors to solve the task to a desired level of accuracy? During each interaction, how much information does an agent need to transmit? These questions are becoming increasingly relevant in the context of edge computing, where limited channel bandwidth and low-power IoT devices (mobile phones, wearable devices etc.) dictate the need for novel strategies to mitigate the communication-bottleneck. In Chapter 6, we drew on ideas from event-triggered control and adaptive quantization to reduce the number of communication rounds, and control the size of the messages being transmitted, in the context of distributed hypothesis testing. One could naturally ask similar questions for other problems on networks. For instance, let us revisit the standard distributed state estimation formulation, where each edge in the underlying graph is now modeled as a discrete-time, noiseless, digital channel that can support a finite number of bits. Even for this simple bit-constrained scenario, the issue of finding the minimal capacity of each channel to solve the distributed estimation problem remains open. Thus, while we have a fair understanding of control/estimation under communication constraints for centralized settings (e.g., stabilizing a plant over a bandwidth-limited channel [171]), questions of the form posed above remain largely unexplored, and offer a ripe avenue for future research at the intersection of network information theory, control systems, and learning theory.

REFERENCES

REFERENCES

[1] S. Park and N. C. Martins, "Design of distributed LTI observers for state omniscience," *IEEE Transactions on Automatic Control*, vol. 62, no. 2, pp. 561–576, 2017.

[2] L. S. Gandin, "Objective analysis of meteorological fields," *Israel Program for Scientific Translations*, vol. 242, 1963.

[3] N. Cressie, "The origins of kriging," *Mathematical Geology*, vol. 22, no. 3, pp. 239–252, 1990.

[4] K. M. Lynch, I. B. Schwartz, P. Yang, and R. A. Freeman, "Decentralized environmental modeling by mobile sensor networks," *IEEE Transactions on Robotics*, vol. 24, no. 3, pp. 710–724, 2008.

[5] R. Graham and J. Cortés, "Adaptive information collection by robotic sensor networks for spatial estimation," *IEEE Transactions on Automatic Control*, vol. 57, no. 6, pp. 1404–1419, 2011.

[6] M. Abazeed, N. Faisal, S. Zubair, and A. Ali, "Routing protocols for wireless multimedia sensor network: a survey," *Journal of Sensors*, 2013.

[7] L. Xie and X. Zhang, "3D clustering-based camera wireless sensor networks for maximizing lifespan with minimum coverage rate constraint," in *Proceedings of the IEEE Global Communications Conference (GLOBECOM)*, 2013, pp. 298–303.

[8] R. N. Smith, M. Schwager, S. L. Smith, B. H. Jones, D. Rus, and G. S. Sukhatme, "Persistent ocean monitoring with underwater gliders: Adapting sampling resolution," *Journal of Field Robotics*, vol. 28, no. 5, pp. 714–741, 2011.

[9] M. Dunbabin, J. M. Roberts, K. Usher, and P. Corke, "A new robot for environmental monitoring on the Great Barrier Reef," in *Proceedings of the Australasian Conference on Robotics & Automation.* Australian Robotics & Automation Association, 2004.

[10] S. Srinivasan, H. Latchman, J. Shea, T. Wong, and J. McNair, "Airborne traffic surveillance systems: video surveillance of highway traffic," in *Proceedings of the ACM 2nd International Workshop on Video surveillance & Sensor Networks.* ACM, 2004, pp. 131–135.

[11] S. L. Smith, M. Schwager, and D. Rus, "Persistent robotic tasks: Monitoring and sweeping in changing environments," *IEEE Transactions on Robotics*, vol. 28, no. 2, pp. 410–426, 2012.

[12] S. Martínez, "Distributed interpolation schemes for field estimation by mobile sensor networks," *IEEE Transactions on Control Systems Technology*, vol. 18, no. 2, pp. 491–500, 2010.

[13] P. Ogren, E. Fiorelli, and N. E. Leonard, "Cooperative control of mobile sensor networks: Adaptive gradient climbing in a distributed environment," *IEEE Transactions on Automatic control*, vol. 49, no. 8, pp. 1292–1302, 2004.

[14] K. Qian, A. Song, J. Bao, and H. Zhang, "Small teleoperated robot for nuclear radiation and chemical leak detection," *International Journal of Advanced Robotic Systems*, vol. 9, no. 3, p. 70, 2012.

[15] A. H. Zakaria, Y. M. Mustafah, J. Abdullah, N. Khair, and T. Abdullah, "Development of autonomous radiation mapping robot," *Procedia Computer Science*, vol. 105, pp. 81–86, 2017.

[16] T. Moore, "Robots for nuclear power plants," *IAEA Bulletin*, vol. 27, no. 3, pp. 31–38, 1985.

[17] L. Wang, A. Morse, D. Fullmer, and J. Liu, "A hybrid observer for a distributed linear system with a changing neighbor graph," in *Proceedings of the 56th IEEE Conference on Decision and Control*, 2017, pp. 1024–1029.

[18] A. V. Banerjee, "A simple model of herd behavior," *The quarterly journal of economics*, vol. 107, no. 3, pp. 797–817, 1992.

[19] S. Bikhchandani, D. Hirshleifer, and I. Welch, "A theory of fads, fashion, custom, and cultural change as informational cascades," *Journal of political Economy*, vol. 100, no. 5, pp. 992–1026, 1992.

[20] I. Lobel, "Social networks: rational learning and information aggregation," ALFRED P SLOAN SCHOOL OF MANAGEMENT CAMBRIDGE MA, Tech. Rep., 2009.

[21] A. Montanari and A. Saberi, "The spread of innovations in social networks," *Proceedings of the National Academy of Sciences*, vol. 107, no. 47, pp. 20 196–20 201, 2010.

[22] D. Acemoglu and A. Ozdaglar, "Opinion dynamics and learning in social networks," *Dynamic Games and Applications*, vol. 1, no. 1, pp. 3–49, 2011.

[23] M. I. Jordan, J. D. Lee, and Y. Yang, "Communication-efficient distributed statistical inference," *Journal of the American Statistical Association*, 2018.

[24] 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.* ACM, 1999, pp. 263–270.

[25] I. F. Akyildiz, W. Su, Y. Sankarasubramaniam, and E. Cayirci, "A survey on sensor networks," *IEEE Communications magazine*, vol. 40, no. 8, pp. 102–114, 2002.

[26] U. Khan, S. Kar, A. Jadbabaie, and J. M. Moura, "On connectivity, observability, and stability in distributed estimation," in *Proceedings of the 49th IEEE Conference on Decision and Control*, 2010, pp. 6639–6644.

[27] U. A. Khan and A. Jadbabaie, "Collaborative scalar-gain estimators for potentially unstable social dynamics with limited communication," *Automatica*, vol. 50, no. 7, pp. 1909–1914, 2014.

[28] S. Park and N. C. Martins, "An augmented observer for the distributed estimation problem for LTI systems," in *Proceedings of the American Control Conference*, 2012, pp. 6775–6780.

[29] ——, "Necessary and sufficient conditions for the stabilizability of a class of LTI distributed observers," in *Proceedings of the 51st IEEE Conference on Decision and Control*, 2012, pp. 7431–7436.

[30] L. Wang and A. S. Morse, "A distributed observer for a time-invariant linear system," *IEEE Transactions on Automatic Control*, vol. 63, no. 7, pp. 2123–2130, 2018.

[31] T. Kim, H. Shim, and D. D. Cho, "Distributed Luenberger observer design," in *Proceedings of the 55th IEEE Conference on Decision and Control*, 2016, pp. 6928–6933.

[32] V. Ugrinovskii, "Conditions for detectability in distributed consensus-based observer networks," *IEEE Transactions on Automatic Control*, vol. 58, no. 10, pp. 2659–2664, 2013.

[33] D. Acemoglu, A. Nedić, and A. Ozdaglar, "Convergence of rule-of-thumb learning rules in social networks," in *Proceedings of the 47th IEEE Conference on Decision and Control*, 2008, pp. 1714–1720.

[34] R. Carli, A. Chiuso, L. Schenato, and S. Zampieri, "Distributed Kalman filtering based on consensus strategies," *IEEE Journal on Selected Areas in Communications*, vol. 26, no. 4, pp. 622–633, 2008.

[35] A. Speranzon, C. Fischione, and K. H. Johansson, "Distributed and collaborative estimation over wireless sensor networks," in *Proceedings of the 45th IEEE Conference on Decision and Control*, 2006, pp. 1025–1030.

[36] J. L. Speyer, "Computation and transmission requirements for a decentralized linear-quadratic-Gaussian control problem," *IEEE Transactions on Automatic Control*, vol. 24, no. 2, pp. 266–269, 1979.

[37] B. Rao, H. Durrant-Whyte, and J. Sheen, "A fully decentralized multi-sensor system for tracking and surveillance," *The International Journal of Robotics Research*, vol. 12, no. 1, pp. 20–44, 1993.

[38] R. Olfati-Saber, "Distributed Kalman filter with embedded consensus filters," in *Proceedings of the 44th IEEE Conference on Decision and Control and European Control Conference*, 2005, pp. 8179–8184.

[39] R. Olfati-Saber and J. S. Shamma, "Consensus filters for sensor networks and distributed sensor fusion," in *Proceedings of the 44th IEEE Conference on Decision and Control and European Control Conference*, 2005, pp. 6698–6703.

[40] R. Olfati-Saber, "Distributed Kalman filtering for sensor networks," in *Proceedings of the 46th IEEE Conference on Decision and Control*, 2007, pp. 5492–5498.

[41] ——, “Kalman-consensus filter: Optimality, stability, and performance,” in *Proceedings of the 48th IEEE Conference on Decision and Control held jointly with the 28th Chinese Control Conference*, 2009, pp. 7036–7042.

[42] M. Kamgarpour and C. Tomlin, “Convergence properties of a decentralized Kalman filter,” in *Proceedings of the 47th IEEE Conference on Decision and Control*, 2008, pp. 3205–3210.

[43] U. Khan and J. M. Moura, “Distributing the Kalman filter for large-scale systems,” *IEEE Transactions on Signal Processing*, vol. 56, no. 10, pp. 4919–4935, 2008.

[44] E. J. Msechu, S. Roumeliotis, A. Ribeiro, and G. B. Giannakis, “Decentralized quantized Kalman filtering with scalable communication cost,” *IEEE Transactions on Signal Processing*, vol. 56, no. 8, pp. 3727–3741, 2008.

[45] U. A. Khan and A. Jadbabaie, “On the stability and optimality of distributed Kalman filters with finite-time data fusion,” in *Proceedings of the American Control Conference*, 2011, pp. 3405–3410.

[46] D. W. Casbeer and R. Beard, “Distributed information filtering using consensus filters,” in *Proceedings of the American Control Conference*, 2009, pp. 1882–1887.

[47] V. Ugrinovskii, “Distributed robust filtering with H_∞ consensus of estimates,” *Automatica*, vol. 47, no. 1, pp. 1–13, 2011.

[48] ——, “Distributed robust estimation over randomly switching networks using H_∞ consensus,” *Automatica*, vol. 49, no. 1, pp. 160–168, 2013.

[49] I. Matei and J. S. Baras, “Consensus-based linear distributed filtering,” *Automatica*, vol. 48, no. 8, pp. 1776–1782, 2012.

[50] B. Shen, Z. Wang, and Y. S. Hung, “Distributed H_∞ - consensus filtering in sensor networks with multiple missing measurements: the finite-horizon case,” *Automatica*, vol. 46, no. 10, pp. 1682–1688, 2010.

[51] D. Ding, Z. Wang, H. Dong, and H. Shu, “Distributed H_∞ state estimation with stochastic parameters and nonlinearities through sensor networks: the finite-horizon case,” *Automatica*, vol. 48, no. 8, pp. 1575–1585, 2012.

[52] P. Millán, L. Orihuela, C. Vivas, and F. R. Rubio, “Distributed consensus-based estimation considering network induced delays and dropouts,” *Automatica*, vol. 48, no. 10, pp. 2726–2729, 2012.

[53] M. Farina, G. Ferrari-Trecate, and R. Scattolini, “Distributed moving horizon estimation for linear constrained systems,” *IEEE Transactions on Automatic Control*, vol. 55, no. 11, pp. 2462–2475, 2010.

[54] V. Ugrinovskii, “Detectability of distributed consensus-based observer networks: An elementary analysis and extensions,” in *Proceedings of the 4th IEEE Australian Control Conference*, 2014, pp. 188–192.

[55] B. D. Anderson and D. J. Clements, “Algebraic characterization of fixed modes in decentralized control,” *Automatica*, vol. 17, no. 5, pp. 703–712, 1981.

[56] E. Davison and Ü. Özgüner, “Characterizations of decentralized fixed modes for interconnected systems,” *Automatica*, vol. 19, no. 2, pp. 169–182, 1983.

[57] A. Mitra and S. Sundaram, “An approach for distributed state estimation of LTI systems,” in *Proceedings of the 2016 54th Annual Allerton Conference on Communication, Control, and Computing*, 2016, pp. 1088–1093.

[58] ——, “Distributed observers for LTI systems,” *IEEE Transactions on Automatic Control*, vol. 63, no. 11, pp. 3689–3704, 2018.

[59] ——, “Distributed functional observers for LTI systems,” in *Proceedings of the 56th IEEE Conference on Decision and Control*, 2017, pp. 3519–3524.

[60] J. A. Bondy and U. S. R. Murty, *Graph Theory with Applications.* Citeseer, 1976, vol. 290.

[61] M. Marcus and H. Minc, *A Survey of Matrix Theory and Matrix Inequalities.* Courier Corporation, 1992, vol. 14.

[62] N. A. Lynch, *Distributed Algorithms.* Morgan Kaufmann, 1996.

[63] M. Darouach, “Existence and design of functional observers for linear systems,” *IEEE Transactions on Automatic Control*, vol. 45, no. 5, pp. 940–943, 2000.

[64] T. L. Fernando, H. M. Trinh, and L. Jennings, “Functional observability and the design of minimum order linear functional observers,” *IEEE Transactions on Automatic Control*, vol. 55, no. 5, pp. 1268–1273, 2010.

[65] W. Y. Leong, H. Trinh, and T. Fernando, “A practical functional observer scheme for interconnected time-delay systems,” *International Journal of Control*, vol. 88, no. 10, pp. 1963–1973, 2015.

[66] T. Fernando, L. Jennings, and H. Trinh, “Functional observability,” in *Proceedings of the 2010 5th International Conference on Information and Automation for Sustainability*, 2010, pp. 421–423.

[67] L. S. Jennings, T. L. Fernando, and H. M. Trinh, “Existence conditions for functional observability from an eigenspace perspective,” *IEEE Transactions on Automatic Control*, vol. 56, no. 12, pp. 2957–2961, 2011.

[68] D. Liberzon, *Switching in systems and control.* Springer Science & Business Media, 2012.

[69] F. Pasqualetti, F. Dörfler, and F. Bullo, “Attack detection and identification in cyber-physical systems,” *IEEE Transactions on Automatic Control*, vol. 58, no. 11, pp. 2715–2729, 2013.

[70] S. Sundaram and C. N. Hadjicostis, “Distributed function calculation via linear iterative strategies in the presence of malicious agents,” *IEEE Transactions on Automatic Control*, vol. 56, no. 7, pp. 1495–1508, 2011.

[71] C.-Z. Bai, F. Pasqualetti, and V. Gupta, “Data-injection attacks in stochastic control systems: Detectability and performance tradeoffs,” *Automatica*, vol. 82, pp. 251–260, 2017.

[72] A. Teixeira, I. Shames, H. Sandberg, and K. H. Johansson, “A secure control framework for resource-limited adversaries,” *Automatica*, vol. 51, pp. 135–148, 2015.

[73] M. Pajic, I. Lee, and G. J. Pappas, “Attack-resilient state estimation for noisy dynamical systems,” *IEEE Transactions on Control of Network Systems*, vol. 4, no. 1, pp. 82–92, 2017.

[74] S. Mishra, Y. Shoukry, N. Karamchandani, S. N. Diggavi, and P. Tabuada, “Secure state estimation against sensor attacks in the presence of noise,” *IEEE Transactions on Control of Network Systems*, vol. 4, no. 1, pp. 49–59, 2017.

[75] Y. Mo and B. Sinopoli, “Secure estimation in the presence of integrity attacks,” *IEEE Transactions on Automatic Control*, vol. 60, no. 4, pp. 1145–1151, 2015.

[76] Y. Chen, S. Kar, and J. M. Moura, “Resilient Distributed Estimation Through Adversary Detection,” *IEEE Transactions on Signal Processing*, 2018.

[77] W. Hashlamoun, S. Brahma, and P. K. Varshney, “Mitigation of Byzantine Attacks on Distributed Detection Systems Using Audit Bits,” *IEEE Transactions on Signal and Information Processing over Networks*, vol. 4, no. 1, pp. 18–32, 2018.

[78] N. Forti, G. Battistelli, L. Chisci, S. Li, B. Wang, and B. Sinopoli, “Distributed joint attack detection and secure state estimation,” *IEEE Transactions on Signal and Information Processing over Networks*, vol. 4, no. 1, pp. 96–110, 2018.

[79] I. Matei, J. S. Baras, and V. Srinivasan, “Trust-based multi-agent filtering for increased smart grid security,” in *Proceedings of the Mediterranean Conference on Control & Automation*, 2012, pp. 716–721.

[80] U. Khan and A. M. Stankovic, “Secure distributed estimation in cyber-physical systems,” in *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing*, 2013, pp. 5209–5213.

[81] M. Deghat, V. Ugrinovskii, I. Shames, and C. Langbort, “Detection of biasing attacks on distributed estimation networks,” in *Proceedings of the IEEE Conference on Decision and Control*, 2016, pp. 2134–2139.

[82] H. Fawzi, P. Tabuada, and S. Diggavi, “Secure estimation and control for cyber-physical systems under adversarial attacks,” *IEEE Transactions on Automatic Control*, vol. 59, no. 6, pp. 1454–1467, 2014.

[83] M. S. Chong, M. Wakaiki, and J. P. Hespanha, “Observability of linear systems under adversarial attacks,” in *Proceedings of the American Control Conference*, 2015, pp. 2439–2444.

[84] A. Mitra and S. Sundaram, “Secure distributed observers for a class of linear time invariant systems in the presence of Byzantine adversaries,” in *Proceedings of the 55th IEEE Conference on Decision and Control*, 2016, pp. 2709–2714.

[85] ——, “Byzantine-resilient distributed observers for LTI systems,” *Automatica*, vol. 108, p. 108487, 2019.

[86] D. Dolev, N. A. Lynch, S. S. Pinter, E. W. Stark, and W. E. Weihl, "Reaching approximate agreement in the presence of faults," *Journal of the ACM (JACM)*, vol. 33, no. 3, pp. 499–516, 1986.

[87] N. H. Vaidya, L. Tseng, and G. Liang, "Iterative approximate Byzantine consensus in arbitrary directed graphs," in *Proceedings of the ACM symposium on Principles of distributed computing*, 2012, pp. 365–374.

[88] H. J. LeBlanc, H. Zhang, X. Koutsoukos, and S. Sundaram, "Resilient asymptotic consensus in robust networks," *IEEE Journal on Selected Areas in Communications*, vol. 31, no. 4, pp. 766–781, 2013.

[89] S. Sundaram and B. Gharesifard, "Distributed optimization under adversarial nodes," *IEEE Transactions on Automatic Control*, vol. 64, no. 3, pp. 1063–1076, 2018.

[90] L. Su and N. H. Vaidya, "Fault-tolerant multi-agent optimization: optimal iterative distributed algorithms," in *Proceedings of the 2016 ACM Symposium on Principles of Distributed Computing.* ACM, 2016, pp. 425–434.

[91] C.-Y. Koo, "Broadcast in radio networks tolerating Byzantine adversarial behavior," in *Proceedings of the twenty-third annual ACM symposium on Principles of distributed computing.* ACM, 2004, pp. 275–282.

[92] A. Pelc and D. Peleg, "Broadcasting with locally bounded Byzantine faults," *Information Processing Letters*, vol. 93, no. 3, pp. 109–115, 2005.

[93] J. Usevitch and D. Panagou, "Resilient leader-follower consensus to arbitrary reference values," in *Proceedings of the Annual American Control Conference.* IEEE, 2018, pp. 1292–1298.

[94] J. Usevitch, K. Garg, and D. Panagou, "Finite-time resilient formation control with bounded inputs," in *Proceedings of the IEEE Conference on Decision and Control*, 2018, pp. 2567–2574.

[95] H. Park and S. A. Hutchinson, "Fault-tolerant rendezvous of multirobot systems," *IEEE Transactions on Robotics*, vol. 33, no. 3, pp. 565–582, 2017.

[96] S. M. Dibaji and H. Ishii, "Resilient consensus of second-order agent networks: Asynchronous update rules with delays," *Automatica*, vol. 81, pp. 123–132, 2017.

[97] A. Mustafa and H. Modares, "Analysis and detection of cyber-physical attacks in distributed sensor networks," in *Proceedings of the Annual Allerton Conference on Communication, Control, and Computing*, 2018, pp. 973–980.

[98] J. Kim, J. G. Lee, C. Lee, H. Shim, and J. H. Seo, "Local identification of sensor attack and distributed resilient state estimation for linear systems," in *Proceedings of the IEEE Conference on Decision and Control*, 2018, pp. 2056–2061.

[99] X. He, X. Ren, H. Sandberg, and K. H. Johansson, "Secure distributed filtering for unstable dynamics under compromised observations," in *Proceedings of the IEEE Conference on Decision and Control*, 2019, pp. 5344–5349.

[100] A. Pagourtzis, G. Panagiotakos, and D. Sakavalas, “Reliable broadcast with respect to topology knowledge,” *Distributed Computing*, vol. 30, no. 2, pp. 87–102, 2017.

[101] F. Pasqualetti, A. Bicchi, and F. Bullo, “Consensus computation in unreliable networks: A system theoretic approach,” *IEEE Transactions on Automatic Control*, vol. 57, no. 1, pp. 90–104, 2012.

[102] W. Han, H. L. Trentelman, Z. Wang, and Y. Shen, “A simple approach to distributed observer design for linear systems,” *IEEE Transactions on Automatic Control*, 2018.

[103] O. Kosut, L. Jia, R. J. Thomas, and L. Tong, “Malicious data attacks on the smart grid,” *IEEE Transactions on Smart Grid*, vol. 2, no. 4, pp. 645–658, 2011.

[104] J. P. Hespanha, *Linear systems theory.* Princeton University Press, 2018.

[105] R. A. Horn, R. A. Horn, and C. R. Johnson, *Matrix analysis.* Cambridge university press, 1990.

[106] H. Zhang and S. Sundaram, “Robustness of information diffusion algorithms to locally bounded adversaries,” in *Proceedings of the American Control Conference*, 2012, pp. 5855–5861.

[107] S. Janson, T. Łuczak, T. Turova, T. Vallier *et al.*, “Bootstrap percolation on the random graph $G_{N,P}$,” *The Annals of Applied Probability*, vol. 22, no. 5, pp. 1989–2047, 2012.

[108] R. Albert and A.-L. Barabási, “Statistical mechanics of complex networks,” *Reviews of modern physics*, vol. 74, no. 1, p. 47, 2002.

[109] P. Erdős and A. Rényi, “On the strength of connectedness of a random graph,” *Acta Mathematica Academiae Scientiarum Hungarica*, vol. 12, no. 1-2, pp. 261–267, 1964.

[110] M. Bradonjić and I. Saniee, “Bootstrap percolation on random geometric graphs,” *Probability in the Engineering and Informational Sciences*, vol. 28, no. 2, pp. 169–181, 2014.

[111] G. J. Pottie and W. J. Kaiser, “Wireless integrated network sensors,” *Communications of the ACM*, vol. 43, no. 5, pp. 51–58, 2000.

[112] C.-T. Chen, *Linear system theory and design.* Oxford University Press, Inc., 1995.

[113] F. F. Rego, A. P. Aguiar, A. M. Pascoal, and C. N. Jones, “A design method for distributed Luenberger observers,” in *Proceedings of the 56th IEEE Conference on Decision and Control*, 2017, pp. 3374 – 3379.

[114] Á. R. del Nozal, P. Millán, L. Orihuela, A. Seuret, and L. Zaccarian, “Distributed estimation based on multi-hop subspace decomposition,” *Automatica*, vol. 99, pp. 213–220, 2019.

[115] S. Wang and W. Ren, “On the convergence conditions of distributed dynamic state estimation using sensor networks: A unified framework,” *IEEE Transactions on Control Systems Technology*, vol. 26, no. 4, pp. 1300–1316, 2018.

[116] J. P. Hespanha, P. Naghshtabrizi, and Y. Xu, "A survey of recent results in networked control systems," *Proceedings of the IEEE*, vol. 95, no. 1, pp. 138–162, 2007.

[117] A. Mitra, J. A. Richards, S. Bagchi, and S. Sundaram, "Resilient distributed state estimation with mobile agents: overcoming Byzantine adversaries, communication losses, and intermittent measurements," *Autonomous Robots*, vol. 43, no. 3, pp. 743–768, 2019.

[118] S. Mou, J. Liu, and A. S. Morse, "A distributed algorithm for solving a linear algebraic equation," *IEEE Transactions on Automatic Control*, vol. 60, no. 11, pp. 2863–2878, 2015.

[119] 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, 2003.

[120] A. Nedić and A. Olshevsky, "Distributed optimization over time-varying directed graphs," *IEEE Transactions on Automatic Control*, vol. 60, no. 3, pp. 601–615, 2014.

[121] H. Lin and P. J. Antsaklis, "Stability and stabilizability of switched linear systems: a survey of recent results," *IEEE Trans. on Autom. control*, vol. 54, no. 2, pp. 308–322, 2009.

[122] S. Kaul, R. Yates, and M. Gruteser, "Real-time status: How often should one update?" in *IEEE INFOCOM*, 2012, pp. 2731–2735.

[123] M. Costa, M. Codreanu, and A. Ephremides, "On the age of information in status update systems with packet management," *IEEE Transactions on Information Theory*, vol. 62, no. 4, pp. 1897–1910, 2016.

[124] R. Talak, S. Karaman, and E. Modiano, "Minimizing age-of-information in multi-hop wireless networks," in *Proceedings Annual Allerton Conf. on Comm., Control, and Computing*, 2017, pp. 486–493.

[125] A. Mitra, J. A. Richards, S. Bagchi, and S. Sundaram, "Finite-time distributed state estimation over time-varying graphs: Exploiting the age-of-information," in *Proc. of the American Control Conference*. IEEE, 2019, pp. 4006–4011.

[126] ——, "Distributed state estimation over time-varying graphs: Exploiting the age-of-information," *arXiv preprint arXiv:2001.07006*, 2020.

[127] A. Jadbabaie, P. Molavi, A. Sandroni, and A. Tahbaz-Salehi, "Non-Bayesian social learning," *Games and Economic Behavior*, vol. 76, no. 1, pp. 210–225, 2012.

[128] A. Jadbabaie, P. Molavi, and A. Tahbaz-Salehi, "Information heterogeneity and the speed of learning in social networks," *Columbia Bus. Sch. Res. Paper*, pp. 13–28, 2013.

[129] P. Molavi, A. Tahbaz-Salehi, and A. Jadbabaie, "A theory of non-Bayesian social learning," *Econometrica*, vol. 86, no. 2, pp. 445–490, 2018.

[130] V. V. Veeravalli, T. Basar, and H. V. Poor, "Decentralized sequential detection with a fusion center performing the sequential test," *IEEE Transactions on Information Theory*, vol. 39, no. 2, pp. 433–442, 1993.

[131] R. Viswanathan and P. K. Varshney, "Distributed detection with multiple sensors Part I. Fundamentals," *Proc. of the IEEE*, vol. 85, no. 1, pp. 54–63, 1997.

[132] J. N. Tsitsiklis, "Decentralized detection by a large number of sensors," *Mathematics of Control, Signals and Systems*, vol. 1, no. 2, pp. 167–182, 1988.

[133] Q. Liu, A. Fang, L. Wang, and X. Wang, "Social learning with time-varying weights," *Journal of Systems Science and Complexity*, vol. 27, no. 3, pp. 581–593, 2014.

[134] H. Salami, B. Ying, and A. H. Sayed, "Social learning over weakly connected graphs," *IEEE Transactions on Signal and Information Processing over Networks*, vol. 3, no. 2, pp. 222–238, 2017.

[135] K. R. Rad and A. Tahbaz-Salehi, "Distributed parameter estimation in networks," in *Proceedings of the 49th IEEE Decision and Control Conference*, 2010, pp. 5050–5055.

[136] S. Shahrampour and A. Jadbabaie, "Exponentially fast parameter estimation in networks using distributed dual averaging," in *Proceedings of the 52nd Decision and Control Conference*, 2013, pp. 6196–6201.

[137] S. Shahrampour, A. Rakhlin, and A. Jadbabaie, "Distributed detection: Finite-time analysis and impact of network topology," *IEEE Transactions on Automatic Control*, vol. 61, no. 11, pp. 3256–3268, 2016.

[138] A. Nedić, A. Olshevsky, and C. A. Uribe, "Fast convergence rates for distributed Non-Bayesian learning," *IEEE Transactions on Automatic Control*, vol. 62, no. 11, pp. 5538–5553, 2017.

[139] ——, "Nonasymptotic convergence rates for cooperative learning over time-varying directed graphs," in *Proceedings of the American Control Conference*. IEEE, 2015, pp. 5884–5889.

[140] A. Lalitha, T. Javidi, and A. Sarwate, "Social learning and distributed hypothesis testing," *IEEE Transactions on Information Theory*, vol. 64, no. 9, pp. 6161–6179, 2018.

[141] A. Lalitha and T. Javidi, "Large deviation analysis for learning rate in distributed hypothesis testing," in *Proceedings of the 49th Asilomar Conference on Signals, Systems and Computers*. IEEE, 2015, pp. 1065–1069.

[142] L. Su and N. H. Vaidya, "Defending Non-Bayesian learning against adversarial attacks," *Distributed Computing*, pp. 1–13, 2016.

[143] R. Olfati-Saber, E. Franco, E. Frazzoli, and J. S. Shamma, "Belief consensus and distributed hypothesis testing in sensor networks," in *Networked Embedded Sens. and Cont.* Springer, 2006, pp. 169–182.

[144] V. Saligrama, M. Alanyali, and O. Savas, "Distributed detection in sensor networks with packet losses and finite capacity links," *IEEE Transactions on Signal Processing*, vol. 54, no. 11, pp. 4118–4132, 2006.

[145] G. L. Gilardoni and M. K. Clayton, "On reaching a consensus using DeGroot's iterative pooling," *The Annals of Stat.*, pp. 391–401, 1993.

[146] A. Mitra, J. A. Richards, and S. Sundaram, "A new approach for distributed hypothesis testing with extensions to Byzantine-resilience," in *Proceedings of the American Control Conference*, 2019, pp. 261–266.

[147] ——, "A new approach to distributed hypothesis testing and non-Bayesian learning: Improved learning rate and Byzantine-resilience," *arXiv preprint arXiv:1907.03588*, 2019.

[148] A. Nedić, A. Olshevsky, and C. A. Uribe, "Distributed learning for cooperative inference," *arXiv preprint arXiv:1704.02718*, 2017.

[149] T. M. Cover and J. A. Thomas, *Elements of Information theory.* John Wiley & Sons, 2012.

[150] P. Molavi, A. Jadbabaie, K. R. Rad, and A. Tahbaz-Salehi, "Reaching consensus with increasing information," *IEEE Journal of Selected Topics in Signal Processing*, vol. 7, no. 2, pp. 358–369, 2013.

[151] D. Acemoglu, A. Ozdaglar, and A. ParandehGheibi, "Spread of (mis) information in social networks," *Games and Economic Behavior*, vol. 70, no. 2, pp. 194–227, 2010.

[152] N. Vaidya, "Matrix representation of iterative approximate Byzantine consensus in directed graphs," *arXiv preprint arXiv:1203.1888*, 2012.

[153] W. Mulzer and D. Werner, "Approximating Tverberg points in linear time for any fixed dimension," *Discrete & Computational Geometry*, vol. 50, no. 2, pp. 520–535, 2013.

[154] H. Royden and P. Fitzpatrick, *Real Analysis.* Prentice Hall, 2010.

[155] W. Hoeffding, "Probability inequalities for sums of bounded random variables," in *The Collected Works of Wassily Hoeffding.* Springer, 1994, pp. 409–426.

[156] C. A. Uribe, J. Z. Hare, L. Kaplan, and A. Jadbabaie, "Non-Bayesian social learning with uncertain models over time-varying directed graphs," *arXiv preprint arXiv:1909.04255*, 2019.

[157] P. Tabuada, "Event-triggered real-time scheduling of stabilizing control tasks," *IEEE Transactions on Automatic Control*, vol. 52, no. 9, pp. 1680–1685, 2007.

[158] W. Heemels, K. H. Johansson, and P. Tabuada, "An introduction to event-triggered and self-triggered control," in *Proceedings of the Conference on Decision and Control.* IEEE, 2012, pp. 3270–3285.

[159] A. Mitra, J. A. Richards, and S. Sundaram, "A communication-efficient algorithm for exponentially fast non-Bayesian learning in networks," in *Proceedings of the Conference on Decision and Control*, 2019, pp. 8347–8352.

[160] A. Mitra, S. Bagchi, and S. Sundaram, "Event-triggered distributed inference," *arXiv preprint arXiv:2004.01302*, 2020.

[161] S. Shahrampour, M. A. Rahimian, and A. Jadbabaie, “Switching to learn,” in *Proceedings of the American Control Conference*, 2015, pp. 2918–2923.

[162] J. Z. Hare, C. A. Uribe, L. M. Kaplan, and A. Jadbabaie, “Communication constrained learning with uncertain models,” in *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing*, 2020, pp. 8609–8613.

[163] A. Olshevsky, I. C. Paschalidis, and A. Spiridonoff, “Fully asynchronous push-sum with growing intercommunication intervals,” in *Proceedings of the American Control Conference*, 2018, pp. 591–596.

[164] K. Tsianos, S. Lawlor, and M. G. Rabbat, “Communication/computation tradeoffs in consensus-based distributed optimization,” in *Advances in Neural Information Processing systems*, 2012, pp. 1943–1951.

[165] G. Lan, S. Lee, and Y. Zhou, “Communication-efficient algorithms for decentralized and stochastic optimization,” *Mathematical Programming*, pp. 1–48, 2017.

[166] A. K. Sahu, D. Jakovetic, and S. Kar, “Credo: A communication-efficient distributed estimation algorithm,” in *Proceedings of the IEEE International Symposium on Information Theory*, 2018, pp. 516–520.

[167] C. Nowzari, E. Garcia, and J. Cortés, “Event-triggered communication and control of networked systems for multi-agent consensus,” *Automatica*, vol. 105, pp. 1–27, 2019.

[168] R. Ahlswede and I. Csiszár, “Hypothesis testing with communication constraints,” *IEEE Transactions on Information Theory*, vol. 32, no. 4, pp. 533–542, 1986.

[169] M. Longo, T. D. Lookabaugh, and R. M. Gray, “Quantization for decentralized hypothesis testing under communication constraints,” *IEEE Transactions on Information Theory*, vol. 36, no. 2, pp. 241–255, 1990.

[170] S. Amari *et al.*, “Statistical inference under multiterminal data compression,” *IEEE Transactions on Information Theory*, vol. 44, no. 6, pp. 2300–2324, 1998.

[171] S. Tatikonda and S. Mitter, “Control under communication constraints,” *IEEE Transactions on Automatic Control*, vol. 49, no. 7, pp. 1056–1068, 2004.

[172] T. T. Doan, S. T. Maguluri, and J. Romberg, “Fast convergence rates of distributed subgradient methods with adaptive quantization,” *arXiv preprint arXiv:1810.13245*, 2018.

[173] C. De Persis and P. Frasca, “Robust self-triggered coordination with ternary controllers,” *IEEE Transactions on Automatic Control*, vol. 58, no. 12, pp. 3024–3038, 2013.

[174] W. H. Heemels, M. Donkers, and A. R. Teel, “Periodic event-triggered control for linear systems,” *IEEE Transactions on Automatic Control*, vol. 58, no. 4, pp. 847–861, 2012.

[175] L. Moreau, "Stability of multiagent systems with time-dependent communication links," *IEEE Transactions on Automatic Control*, vol. 50, no. 2, pp. 169–182, 2005.

[176] J. Li, G. Chen, Z. Wu, and X. He, "Distributed subgradient method for multiagent optimization with quantized communication," *Mathematical Methods in the Applied Sciences*, vol. 40, no. 4, pp. 1201–1213, 2017.

[177] M. El Chamie, J. Liu, and T. Başar, "Design and analysis of distributed averaging with quantized communication," *IEEE Transactions on Automatic Control*, vol. 61, no. 12, pp. 3870–3884, 2016.

[178] A. Nedic, A. Olshevsky, A. Ozdaglar, and J. N. Tsitsiklis, "Distributed subgradient methods and quantization effects," in *Proceedings of the 47th IEEE Conference on Decision and Control*, 2008, pp. 4177–4184.

[179] S. Battilotti and M. Mekhail, "Distributed estimation for nonlinear systems," *Automatica*, vol. 107, pp. 562–573, 2019.

[180] A. Chakrabarty, S. Sundaram, M. J. Corless, G. T. Buzzard, S. H. Żak, and A. E. Rundell, "Distributed unknown input observers for interconnected nonlinear systems," in *Proceedings of the American Control Conference (ACC)*, 2016, pp. 101–106.

[181] A. Mitra and S. Sundaram, "A novel switched linear observer for estimating the state of a dynamical process with a mobile agent," in *Proceedings of the 57th IEEE Conference on Decision and Control*, 2018.

[182] L. Wang, D. Fullmer, F. Liu, and A. Morse, "Distributed control of linear multi-variable systems," *arXiv preprint arXiv:1909.11823*, 2019.

[183] P. Landgren, V. Srivastava, and N. E. Leonard, "Distributed cooperative decision making in multi-agent multi-armed bandits," *arXiv preprint arXiv:2003.01312*, 2020.

[184] S. Shahrampour and A. Jadbabaie, "Distributed online optimization in dynamic environments using mirror descent," *IEEE Transactions on Automatic Control*, vol. 63, no. 3, pp. 714–725, 2017.

[185] T. Doan, S. Maguluri, and J. Romberg, "Finite-time analysis of distributed td (0) with linear function approximation on multi-agent reinforcement learning," in *International Conference on Machine Learning*, 2019, pp. 1626–1635.

VITA

VITA

Aritra Mitra was born on October 1st, 1990, in the beautiful city of Calcutta (Kolkata), India. He completed his schooling from Don Bosco Park Circus in 2009, and went on to receive the B.E. degree from Jadavpur University, India, in 2013, and the M.Tech degree from the Indian Institute of Technology, Kanpur, India, in 2015, both in Electrical Engineering. He is currently pursuing a Ph.D. at the School of Electrical and Computer Engineering, Purdue University. His research interests include the study of dynamical processes on networks, multi-agent control and estimation, statistical inference, and the security of cyber-physical systems. He was a recipient of the University Gold Medal at Jadavpur University, the Academic Excellence Award at IIT Kanpur, and the ECE Fellowship at Purdue University. He will join the University of Pennsylvania as a postdoctoral researcher after the completion of this Ph.D.

In his free time, Aritra loves to read mystery novels (the “little grey cells” of Hercule Poirot never cease to amaze him), watch cricket, listen to Indian music, and try his hand out at cooking different cuisines.