8-1-2016

Distributed Inference and Learning with Byzantine Data

Bhavya Kailkhura
Syracuse University

ABSTRACT

We are living in an increasingly networked world with sensing networks of varying shapes and sizes: the network often comprises of several tiny devices (or nodes) communicating with each other via different topologies. To make the problem even more complicated, the nodes in the network can be unreliable due to a variety of reasons: noise, faults and attacks, thus, providing corrupted data. Although the area of statistical inference has been an active area of research in the past, distributed learning and inference in a networked setup with potentially unreliable components has only gained attention recently. The emergence of big and dirty data era demands new distributed learning and inference solutions to tackle the problem of inference with corrupted data.

Distributed inference networks (DINs) consist of a group of networked entities which acquire observations regarding a phenomenon of interest (POI), collaborate with other entities in the network by sharing their inference via different topologies to make a global inference. The central goal of this thesis is to analyze the effect of corrupted (or falsified) data on the inference performance of DINs and design robust strategies to ensure reliable overall performance for several practical network architectures. Specifically, the inference (or learning) process can be that of detection or estimation or classification, and the topology of the system can be parallel, hierarchical or fully decentralized (peer to peer).

Note that, the corrupted data model may seem similar to the scenario where local decisions are transmitted over a Binary Symmetric Channel (BSC) with a certain cross over probability, however, there are fundamental differences. Over the last three decades, research community has extensively studied the impact of transmission channels or faults on the distributed detection system and related problems due to its importance in several applications. However, corrupted (Byzantine) data models considered in this thesis, are philosophically different from the BSC or the faulty sensor cases. Byzantines are intentional and intelligent, therefore, they can optimize over the data corruption parameters. Thus, in contrast to channel aware detection, both the FC and

the Byzantines can optimize their utility by choosing their actions based on the knowledge of their opponent's behavior. Study of these practically motivated scenarios in the presence of Byzantines is of utmost importance, and is missing from the channel aware detection and fault tolerant detection literature. This thesis advances the distributed inference literature by providing fundamental limits of distributed inference with Byzantine data and provides optimal counter-measures (using the insights provided by these fundamental limits) from a network designer's perspective. Note that, the analysis of problems related to strategical interaction between Byzantines and network designed is very challenging (NP-hard is many cases). However, we show that by utilizing the properties of the network architecture, efficient solutions can be obtained. Specifically, we found that several problems related to the design of optimal counter-measures in the inference context are, in fact, special cases of these NP-hard problems which can be solved in polynomial time.

First, we consider the problem of distributed Bayesian detection in the presence of data falsification (or Byzantine) attacks in the parallel topology. Byzantines considered in this thesis are those nodes that are compromised and reprogrammed by an adversary to transmit false information to a centralized fusion center (FC) to degrade detection performance. We show that above a certain fraction of Byzantine attackers in the network, the detection scheme becomes completely incapable (or blind) of utilizing the sensor data for detection. When the fraction of Byzantines is not sufficient to blind the FC, we also provide closed form expressions for the optimal attacking strategies for the Byzantines that most degrade the detection performance. Optimal attacking strategies in certain cases have the minimax property and, therefore, the knowledge of these strategies has practical significance and can be used to implement a robust detector at the FC.

In several practical situations, parallel topology cannot be implemented due to limiting factors, such as, the FC being outside the communication range of the nodes and limited energy budget of the nodes. In such scenarios, a multi-hop network is employed, where nodes are organized hierarchically into multiple levels (tree networks). Next, we study the problem of distributed inference in tree topologies in the presence of Byzantines under several practical scenarios. We analytically characterize the effect of Byzantines on the inference performance of the system. We also look at

the possible counter-measures from the FC's perspective to protect the network from these Byzantines. These counter-measures are of two kinds: Byzantine identification schemes and Byzantine tolerant schemes. Using learning based techniques, Byzantine identification schemes are designed that learn the identity of Byzantines in the network and use this information to improve system performance. For scenarios where this is not possible, Byzantine tolerant schemes, which use game theory and error-correcting codes, are developed that tolerate the effect of Byzantines while maintaining a reasonably good inference performance in the network.

Going a step further, we also consider scenarios where a centralized FC is not available. In such scenarios, a solution is to employ detection approaches which are based on fully distributed consensus algorithms, where all of the nodes exchange information only with their neighbors. For such networks, we analytically characterize the negative effect of Byzantines on the steady-state and transient detection performance of conventional consensus-based detection schemes. To avoid performance deterioration, we propose a distributed weighted average consensus algorithm that is robust to Byzantine attacks. Next, we exploit the statistical distribution of the nodes' data to devise techniques for mitigating the influence of data falsifying Byzantines on the distributed detection system. Since some parameters of the statistical distribution of the nodes' data might not be known a priori, we propose learning based techniques to enable an adaptive design of the local fusion or update rules.

The above considerations highlight the negative effect of the corrupted data on the inference performance. However, it is possible for a system designer to utilize the corrupted data for network's benefit. Finally, we consider the problem of detecting a high dimensional signal based on compressed measurements with secrecy guarantees. We consider a scenario where the network operates in the presence of an eavesdropper who wants to discover the state of the nature being monitored by the system. To keep the data secret from the eavesdropper, we propose to use cooperating trustworthy nodes that assist the FC by injecting corrupted data in the system to deceive the eavesdropper. We also design the system by determining the optimal values of parameters which maximize the detection performance at the FC while ensuring perfect secrecy at the eavesdropper.

DISTRIBUTED INFERENCE AND LEARNING WITH BYZANTINE DATA

By

Bhavya Kailkhura
B.E., Nagpur University, 2010
M.S., Syracuse University, 2012

DISSERTATION

Submitted in partial fulfillment of the requirements for the degree of
Doctor of Philosophy in Electrical and Computer Engineering

Syracuse University
August 2016

ACKNOWLEDGMENTS

"If I have seen further, it is by standing on the shoulders of giants."

Isaac Newton

I am very grateful to my PhD advisor Prof. Pramod K. Varshney for his invaluable guidance throughout this dissertation. He gave me the freedom to select research problems that I believed are important and interesting. He also encouraged me to collaborate with right set of people to work on those problems. Thank you Prof. Varshney for believing in me. I would like to extend my heartfelt thanks to Prof. Yunghsiang Han, for being an excellent mentor to me during my PhD. I have learnt quite a lot from him during my graduate studies, especially the initial stages, which shaped me into who I am now. In addition, I would like to thank my defense committee members Prof. Yingbin Liang, Prof. Mustafa Gursoy, Prof. Pinyun Chen, Prof. Lixin Shen, and Prof. Jian Tang.

I would also like to thank Dr. Peer-Timo Bremer and Dr. Jayaraman J. Thiagarajan for their mentoring during my summer internship at Lawrence Livermore National Laboratory. Besides their guidance on my project, I received a great deal of valuable career advice from them. I have had the pleasure to extensively collaborate with Dr. Thiagarajan during the later half of my PhD to explore multiple research directions. His enthusiasm and creativity is inspiring.

I have had the pleasure of collaborating with multiple researchers during the course of my PhD. I would like to thank my collaborators Aditya, Jay, Karthi, Prashant and Lakshmi. I had many productive and learning experiences working with each of them.

I have had several technical and philosophical discussions about distributed inference and research in general with Dr. Aditya Vempaty and Dr. Arun Subramanian. A big thanks to both of them.

I have enjoyed the company and support of my academic family: Swarnendu, Arun, Sid, Yujiao, Raghed, Hao, Aditya, Sijia, Nianxia, Prashant, Shan, Swatantra, Qunwei, Pranay, Swastik, Sora, and Thakshila. Thanks to my officemates Hao, Nancy, Raghed and Shan for lots of fun and will be greatly missed. Further, I cannot stress enough the importance of the support of my friends outside this academic circle.

No amount of "thank yous" will suffice to express my gratitude to my parents, grandparents, brother and rest of my family for their unconditional love, support and sacrifices throughout my life. It goes without saying that everything that I have achieved in my life, including this dissertation, would not have been possible without them. Thank you ma, papa for being such a great role model and to show me how to be a kind, compassionate and good person. I hope one day I am at least half the amazing you are.

To my family.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

CHAPTER 1
INTRODUCTION

1.1 Distributed Inference Networks

Distributed inference networks (DINs) have attracted much recent attention due to a variety of applications in civilian and military domains. These include distributed spectrum sensing (DSS), traffic and environment monitoring, medical monitoring, power networks, localization and surveillance, etc. DINs employ a group of spatially distributed sensing entities that collaborate to sense and make inferences about a given phenomenon of interest (POI). In the traditional framework of *centralized* inference networks, nodes transmit raw observations to a fusion center (FC) where a global inference is made. These transmissions are not attractive in practice as raw observations require a large bandwidth (or energy) for reliable reception at the FC. Therefore, in DINs, the nodes transmit compressed summaries which are obtained by processing original observations (e.g., using a quantizer or other "transmission function") prior to transmission to the FC. The FC uses a fusion rule to integrate the received information/data to make a global inference about the POI.

1.2 Motivation

In DINs, a large number of inexpensive and less reliable nodes that can provide dense coverage are used to provide a balance between cost and functionality. The inference performance of such systems strongly depends on the reliability of nodes in the network. Further, the distributed nature of such systems makes them quite vulnerable to different types of attacks. Designing robust inference systems against attacks is of utmost importance. While there are several practical challenges one faces while inferring based on observations/information from multiple nodes in DINs, this thesis focuses on the class of challenges associated with the presence of corrupted information. We focus on the causes of corrupted data from the nodes and, without any loss of generality, refer to such nodes as adversaries. These causes can be divided into three types: $1)$ channel noise, $2)$ faulty sensors, and $3)$ malicious attackers. The major focus of this thesis is on the cause (3) where an adversary intentionally injects corrupted data to degrade the inference performance of the system.

In recent years, security issues of such distributed networks are increasingly being studied within the networking, signal processing and information theory communities. In general, there are two kinds of attacks in an inference network: $1)$ active attacks, and $2)$ passive attacks. An active attack is a network exploit in which an adversary attempts to make changes to data to degrade the system performance. On the other hand, in passive attacks, an unauthorized party monitors the network and sometimes observes the transmission of the authorized nodes. The purpose of such eavesdroppers is to gain information about the POI without modifying any data. Note that, passive attacks are often activities in preparation for active attacks.

One typical active attack on distributed inference networks is a Byzantine attack. While Byzantine attacks (originally proposed by [58]) may, in general, refer to many types of malicious behavior, our focus in this thesis is on data corruption attacks. In this type of attack, an attacker may send corrupt (erroneous) data to the FC to degrade inference performance. In this thesis, we refer to such a data corruption attacker as a Byzantine and the data thus generated is referred to as Byzantine data. As an application to distributed detection with Byzantine data, consider the problem of distributed spectrum sensing when some participants attack a cognitive radio network (CRN) by

sending falsified data to the FC. In this context, Byzantine nodes can affect decisions at the FC by reporting false data. This might result in a collision of secondary users with the primary user (PU) (if a busy PU is wrongly detected as idle) or in spectrum wastage (if an idle PU is detected as busy). For distributed estimation, consider the impact of Byzantine data on the state estimation in power grids. Here an adversary may take control of some of the meters and launch a man-in-the-middle (MiM) attack by substituting actual measurements with falsified data. If undetected, state estimates at the FC will be altered and subsequent decisions using state estimates are affected.

Similarly, a typical passive attack on DINs is an eavesdropping attack. The motive of an eavesdropper (Eve) is to compromise the secrecy of a given inference network. For instance, some of the nodes within a cognitive radio network (CRN) may take advantage of the FC's inferences and may compete against the CRN in using the primary user's channels without paying any participation costs to the network moderator. Although the presence of malicious nodes is the focus of this thesis, there are other related security challenges such as jamming and Sybil attacks that are relevant to the problems considered in this thesis but are not considered here.

1.3 Literature Survey

Detection, classification, or estimation of certain events, targets, or phenomena, in a region of interest, is an important application of inference networks. In a conventional parallel topology framework, the objective is usually to find efficient quantization rules for the nodes and efficient inference rule for the FC, which maximize the global performance at the FC (for a comprehensive survey, see [105] and references therein). Several aspects of such a framework are studied [105]: network topology, decision rules, effect of wireless channels, effect of spatio-temporal dependence, etc. Most of the initial work focused on the design of local sensor decision rules and optimal fusion rule at the FC [9, 17, 52, 57, 65, 100, 104, 110, 113, 115]. The advancement of wireless sensor networks (WSNs) renewed interest in this area along with new research challenges: wireless channels [13, 15], network topologies [2, 96], sensor resource management [4, 5, 43, 83], correlated

observations [12, 24, 41, 95], etc. The effect of wireless channels can be addressed by analyzing the system under the channel-aware formulation [15, 76]. Note that, in general, the problem of designing optimal inference rules is computationally expensive (NP-hard) [102]. However, in a distributed detection framework, under the assumption of conditional independence, the optimal decision rule for each node takes the form of a likelihood ratio test with a suitably chosen threshold [14]. Further, it has been shown that the use of identical thresholds is asymptotically optimal [103]. Under the assumption of identical thresholds, several authors have considered the problem of designing optimal decision rules in the past [92, 125].

In contrast to the distributed detection problem, in a classification problem, each decision is usually represented by $\log_2 M$ information bits, where M is the number of classes to be distinguished. The problem of classification using $\log_2 M$ information bits has been studied for parallel topology [6]. Due to bandwidth constraints, it is desirable that the local node decisions are sent to the FC with as few bits as possible. To overcome this problem, distributed classification has been proposed in which the local nodes make 1-bit (rather than $\log_2 M$ bit) local decisions and send them to the FC [117, 126, 129]. The FC then uses the local decisions collectively and makes a global inference about the underlying phenomenon.

In [39, 90], the authors consider the problem of parameter estimation in a parallel topology. Received signal strength based methods have been proposed which employ least-squares or maximum likelihood (ML) based parameter estimation techniques. These techniques are not suitable for power and bandwidth constrained networks. To overcome these drawbacks, distributed parameter estimation using quantized measurements has been addressed in [73, 86, 87]. Similar to the problem of distributed detection, system design issues of distributed estimation have also been addressed only in certain scenarios, such as in [109], where it has been shown that identical quantizers are optimal under certain conditions. In [107], coding theory based iterative schemes were proposed for target localization using a parallel topology where at every iteration, the FC solves an M-ary hypothesis testing problem and decides the region of interest for the next iteration. There also have been limited attempts to address distributed inference problems in tree

networks [30, 47, 48, 99, 128]. In all but the simplest cases, optimal strategies in tree based networks are difficult to derive. Most of the work on tree networks focuses on person-by-person optimal (PBPO) strategies [47, 48, 99, 128].

While there is vast literature on distributed inference, work reported on distributed inference with Byzantine data is still limited, and is the focus of this thesis. This research problem is motivated from the popular Byzantine generals problem [58] as discussed next.

1.3.1 Byzantine Generals Problem

In 1982, Lamport et al. presented the so-called *Byzantine generals problem* as follows [58]: "a group of generals of the Byzantine army camped with their troops around an enemy city. Communicating only by messengers, the generals must agree upon a common battle plan. However, one or more of them may be traitors who will try to mislead the others. The problem is to find an algorithm to ensure that the loyal generals will reach agreement." This problem is similar in principle to the problem considered in this thesis. The authors [58] gave a sharp characterization of the power of the Byzantine generals. It was shown that if the fraction of Byzantine generals is less than $1/3$, there is a way for the loyal generals to reach a consensus agreement, regardless of what the Byzantine generals do. If the fraction is above $1/3$, consensus can no longer be guaranteed. There are many diverse behaviors that a Byzantine entity may engage in, such as a sensor may lie about connectivity, flood network with false traffic, attempt to subjugate control information, falsely describe opinions of another node (e.g., peer to peer), or capture a strategic subset of devices and collude.

Next, we review the Byzantine generals problem in the context of distributed inference in different topologies [112]. Researchers have typically focused on two basic parts of this problem. In the first set of works, the problem is analyzed from the attacker's perspective and optimal attack strategies are derived that result in deterioration of the network's performance [49, 51, 66, 70, 85, 111]. They have modeled the potential attack strategies and optimized over the attack space to determine the optimal attack by the adversary. The second set of works focused on analysis from

the network's perspective to determine the counter-attack strategies to protect the network from these Byzantine attacks [19, 32, 37, 85, 106, 111].

1.3.2 Distributed Inference with Byzantines in Parallel Topologies

Although distributed detection has been a very active field of research in the past, security problems in distributed detection networks have gained attention only very recently. In [66], the authors considered the problem of distributed detection in the presence of Byzantines for a parallel topology under the Neyman-Pearson (NP) setup and determined the optimal attacking strategy which minimizes the detection error exponent. This approach based on Kullback-Leibler divergence (KLD) is analytically tractable and yields approximate results in non-asymptotic cases. They also assumed that the Byzantines know the true hypothesis, which obviously is not satisfied in practice but does provide a bound. In [85], the authors analyzed the same problem in the context of collaborative spectrum sensing under Byzantine Attacks. They relaxed the assumption of perfect knowledge of the hypotheses by assuming that the Byzantines determine the knowledge about the true hypotheses from their own sensing observations. Schemes for Byzantine node identification in parallel topology have been proposed in [19, 85, 93, 106].

Note that, the Byzantine attack model is similar to the scenario where local decisions are transmitted over a Binary Symmetric Channel (BSC) with a certain cross over probability. There are several papers that address the impact of transmission channels or faults on the distributed detection system and related problems [15, 22, 64, 116]. However, Byzantine attacks are philosophically different from the BSC or the faulty sensor case. Byzantine attacks are intentional and, therefore, the attacker can optimize over the attack parameters. Thus, in contrast to channel aware detection, both the FC and the Byzantines can optimize their utility by choosing their actions based on the knowledge of their opponent's behavior. Study of these practically motivated scenarios in the presence of Byzantines is missing from the channel aware detection and fault tolerant detection literature because of the philosophical difference between these approaches.

All the approaches discussed so far consider distributed detection with Byzantines in an NP

setup for parallel topologies. To the best of our knowledge, the problem of distributed Bayesian detection with Byzantines has not been considered in the past. Further, the problem of distributed inference in tree networks with Byzantines has not received any attention. Due to the complexity of inference in tree networks as compared to the parallel topology, these problems have been left unexplored by researchers. In this thesis, we take some first steps toward addressing these problems.

1.3.3 Distributed Inference with Byzantines in Decentralized Topologies

Thus far, research on detection in the presence of Byzantine attacks has predominantly focused on addressing these attacks under the centralized model in which information is available at the FC [66, 69, 85, 106]. A few attempts have been made to address the security threats in "average" consensus-based detection schemes in recent research [59, 62, 97, 121–123]. Most of these existing works on countering Byzantine or data falsification attacks in distributed networks rely on a threshold for detecting Byzantines. The main idea is to exclude nodes from the neighbors list whose state information deviates significantly from the mean value. In [123] and [121], two different defense schemes against data falsification attacks for distributed consensus-based detection were proposed. In [123], the scheme eliminates the state value with the largest deviation from the local mean at each iteration step and, therefore, it can only deal with the situation in which only one Byzantine node exists. Note that, it excludes one state value even if there is no Byzantine node. In [121], the vulnerability of distributed consensus-based spectrum sensing was analyzed and an outlier detection algorithm with an adaptive threshold was proposed. The authors in [62] proposed a Byzantine mitigation technique based on adaptive local thresholds. This scheme mitigates the misbehavior of Byzantine nodes and tolerates the occasional large deviation introduced by honest users. It adaptively reduces the corresponding coefficients so that the Byzantines are eventually isolated from the network.

Excluding the Byzantine nodes from the fusion process may not be the best strategy from the

network's perspective. As shown in [106] in the context of distributed detection with one-bit measurements under a centralized model with an FC, an intelligent way to improve the performance of the network is to use the information of the identified Byzantines to the network's benefit. More specifically, learning-based techniques have the potential to outperform the existing exclusion-based techniques. In this thesis, we pursue such a design philosophy in the context of raw data fusion in decentralized networks based on weighted average consensus algorithms. Byzantines can attack weighted average consensus algorithms in two ways: $1)$ nodes falsify their initial data, and $2)$ nodes falsify their weight values. To the best of our knowledge, the susceptibility and protection of weighted average consensus-based detection schemes has not been considered in the literature. In this thesis, we take some first steps toward addressing the problem of robust consensus-based detection in the presence of Byzantine attacks.

1.4 Research Methodology

Analysis of networked inference systems is much more challenging compared to classical inference systems which ignore how the data is generated. In a distributed inference network both the adversary and the FC have several degrees of freedom which makes the problem complicated. In this thesis, we study the problem of distributed inference with Byzantine data from both attacker's and network designer's perspective. Note that, finding the optimal attacking strategies is the first step toward designing a robust distributed detection system in a holistic manner.

1.4.1 Analysis from Adversary's Perspective

From Byzantines' perspective, one problem of interest is to characterize the degree to which they can affect the inference performance of DINs. We are interested in the minimum fraction of Byzantine nodes that make the detection no better than merely based on prior information without using any data referred to as blinding the network. We refer to the fraction of Byzantines in the network as the attack power of the Byzantines and minimum power to blind the FC as critical power. Some

important questions from a Byzantines' perspective to answer are:

- How does network topology affect this critical power? In other words, which network architectures are more susceptible to Byzantine attacks?
- Can we analytically characterize the critical power in different networks?
- If the power of the adversary is less than this critical value, what should be the optimal attacking strategy of the adversary?
- Further, in a heterogeneous inference network, which resources/nodes should a cost constrained adversary attack to maximize its profit?

While these questions are difficult to answer in general, some insights can be obtained by utilizing tools from hypothesis testing, game theory, information theory, and machine learning.

1.4.2 Analysis from Network Designer's Perspective

The previous discussion addresses the issue of inference from corrupted data from the adversary's perspective. However, one needs to look at the possible countermeasures from the network designer's perspective to protect the network from these Byzantines. We follow the methodology suggested by Claude Shannon in his unpublished manuscript of 1956 titled "Reliable Machines from Unreliable Components" [89] which considers the problem of designing reliable machines from unreliable components. He suggests that there are typically three methods to improve system reliability: $1)$ improve individual system components, $2)$ use of error-correction codes, and $3)$ complete system redesign. As seen later in the thesis, complete redesign corresponds to a total change in the inference architecture. Problem of optimal network structure design which is robust to Byzantines is solved in a distributed inference context. System components are improved either by identifying the local malicious nodes and using them for further inference and/or improving the performance of global detector by implementing the optimal fusion rule. Although the cause

of unreliable information is different for faulty sensors and Byzantines, the effect is the same: errors in the data. Therefore, coding-theory ideas are used to correct these errors and improve the inference performance.

1.5 Outline of the Thesis and Contributions

The central goal of this thesis is to analyze the performance of distributed inference networks with potentially corrupted data and design strategies to ensure reliable inference performance for several practical network topologies. The inference process can involve detection or estimation or classification, and the topology of the system can be parallel, hierarchical or fully decentralized (peer to peer). The thesis is divided in two parts. In the first part, we study the negative effect of corrupted data on inference performance and propose some counter-measures. In the second part of the thesis, we show the positive effect of the friendly data corruption on secrecy performance of the system. We design schemes to intelligently use the corrupted data to improve the secrecy performance of DINs.

An overview of the general model is provided in Chapter 2; literature review and background material needed for the later chapters of the thesis is also presented. In the first part of the thesis, we study the problem of distributed inference with corrupted data for different topologies such as parallel topology (Chapter 3), tree topology (Chapters 4, 5 and 6), and peer to peer topology (Chapter 7). Next, in the second part of the thesis, we look into the positive effect of corrupted data on the secrecy performance of inference networks in Chapter 8. For each of these chapters, we follow the research methodology described in Sec. 1.4. First, we study the effect of corrupted data on inference performance to determine their impact on the overall performance of the system. Then, using these insights, we propose efficient counter-measures and design robust inference systems by $1)$ learning and using Byzantines' parameters, $2)$ using error-correction codes, and $3)$ redesigning network architecture. Finally, the thesis is concluded in Chapter 9 with a summary of results presented in this thesis and future research directions.

Chapter 2: Background

The general system model considered in this thesis is described in Chapter 2 and the taxonomy corresponding to this generalized model is presented. Literature corresponding to this structure is reviewed. Some specific tools used in the development of this thesis are also discussed in this chapter.

Chapter 3: Distributed Detection with Byzantine Data: Parallel Topology

In this chapter, we consider the case of a parallel network performing a detection task using binary quantized data. We consider malicious sensors called Byzantines and investigate the distributed detection problem under a Bayesian framework. The problem of distributed detection is formulated as a binary hypothesis test at the FC based on 1-bit data sent by the sensors. The expression for minimum attacking power required by the Byzantine is derived. We analyze the problem under different practical scenarios where the FC and the Byzantines may or may not have knowledge of their opponent's strategies and derive results for both asymptotic and non-asymptotic cases. It is found that asymptotics based results do not hold under several non-asymptotic scenarios.

In several practical situations, a parallel topology cannot be implemented due to several factors, such as, the FC being outside the communication range of the nodes and limited energy budget of the nodes. In such cases, a multi-hop network is employed, where nodes are organized hierarchically into multiple levels (tree networks). Next, in this thesis, we study the problem of distributed inference in tree topologies in the presence of Byzantines under several practical scenarios. We analytically characterize the effect of Byzantines on the inference performance of the system.

Chapter 4: Distributed Detection with Unlabeled Byzantine Data: Tree Topology

In this chapter, we consider the problem of distributed detection in tree topologies in the presence of Byzantines. It is assumed that the packet IDs (or source IDs) are not forwarded in the tree to save energy (unlabeled data). We show that when more than a certain fraction of individual node decisions are falsified, the decision fusion scheme becomes completely incapable. We also

look at the possible counter-measures from the FC's perspective to protect the network from these Byzantines. We formulate the robust topology design problem as a bi-level program and provide an efficient algorithm to solve it, which is guaranteed to find an optimal solution, if one exists.

Chapter 5: Distributed Detection with Labeled Byzantine Data: Tree Topology

Similar to Chapter 4, the problem of distributed detection in tree networks in the presence of Byzantines is considered. However, the assumption of unlabeled data is relaxed, thus, the FC is aware of the source IDs. In such scenarios, closed form expressions for optimal attacking strategies that minimize the miss detection error exponent at the FC are obtained. Further, we study the problem of designing optimal distributed detection parameters. Next, we model the strategic interaction between the FC and the attacker as a Leader-Follower (Stackelberg) game. This formulation provides a methodology for predicting attacker and defender (FC) equilibrium strategies, which is used to implement the optimal detector. Finally, a reputation based scheme to identify Byzantines is proposed and its performance is analytically evaluated.

Chapter 6: Distributed Inference in the Presence of Faults: Tree Topology

In the framework considered in this chapter, distributed nodes make a 1-bit local decision regarding a phenomenon before sending it to the FC via intermediate nodes. We propose the use of coding theory based techniques to solve fault tolerant distributed inference problem in such structures. Data fusion at nodes as well as at the FC is implemented via error correcting codes. In this context, we analyze the performance for a given code matrix and also design the optimal code matrices at every level of the tree. We address the problems of distributed classification and distributed estimation separately and develop schemes to perform these tasks in tree networks. We show that the proposed schemes are asymptotically optimal under certain conditions. Fault-tolerance capability of the scheme is verified using simulation results.

Note that, the detection schemes considered above rely on a centralized FC where a global decision is made. However, in many scenarios, a FC may not be available. Going a step further,

we next consider the problem of distributed detection with Byzantines in peer to peer network.

Chapter 7: Distributed Detection with Byzantine Data: Peer to Peer Topology

This chapter considers the problem of signal detection in distributed networks in the presence of data falsification (Byzantine) attacks in peer to peer networks. Detection approaches considered in the paper are based on fully distributed consensus algorithms, where all of the nodes exchange information only with their neighbors in the absence of a fusion center. For such networks, we first characterize the negative effect of Byzantines on the steady-state and transient detection performance of conventional consensus-based detection algorithms. To avoid performance deterioration, we propose a distributed weighted average consensus algorithm that is robust to Byzantine attacks. We show that, under reasonable assumptions, the global test statistic for detection can be computed locally at each node using our proposed consensus algorithm. Next, we exploit the statistical distribution of the nodes' data to devise techniques for mitigating the influence of data falsifying Byzantines on the distributed detection system. Since some parameters of the statistical distribution of the nodes' data might not be known a priori, we propose learning based techniques to enable an adaptive design of the local fusion or update rules. Our scheme differs from (and outperforms) all existing work on Byzantine mitigation that are based on exclusion strategies [62, 97, 121–123], where the only defense is to identify and exclude the attackers from the consensus process.

Previous chapters highlight the negative effect of corrupted data or data falsification on the inference performance of the system. However, it is possible for a system designer to utilize the corrupted data for network's benefit. Motivated from this fact, in Chapter 8, we study the positive use of the falsified data to improve the secrecy performance of the system.

Chapter 8: Compressive Detection with an Eavesdropper: Exploiting Corrupted data to Improve Secrecy Performance

In this chapter, we consider the problem of detecting a high dimensional signal based on compressed measurements with physical layer secrecy guarantees. First, we propose a collaborative

compressive detection (CCD) framework to compensate for the performance loss due to compression with a single sensor. Next, we consider a scenario where the network operates in the presence of an eavesdropper who wants to discover the state of nature being monitored by the system. To keep the data secret from the eavesdropper, we propose to use cooperating trustworthy nodes that assist the FC by injecting corrupted data to deceive the eavesdropper. Further, we design optimal measurement matrices to obtain compressed data at distributed nodes so that the detection performance of the network is maximized while guaranteeing a certain level of secrecy. We solve the measurement matrix design problem for three different scenarios: $a)$ the signal is known, $b)$ the signal lies in a low dimensional subspace, and $c)$ the signal is sparse. We show that the secrecy performance of the system can be improved by using optimized measurement matrices along with artificial noise injection based techniques.

Chapter 9: Conclusion

In this chapter, we first recapitulate the main ideas and results presented in the thesis. Then some directions for extending the thesis are given that are derived from the general formulations of problems and solution methodologies presented in the thesis.

1.6 Summary of Contributions

Tables 1.1 and 1.2 summarizes the contributions of the thesis. The contributions of the thesis can be split into two broad parts: $1)$ derivation of fundamental limits of distributed inference with Byzantine data in parallel, tree and peer to peer topologies, and $2)$ design of optimal countermeasures using the insights provided by these fundamental limits.

Bibliographic Note

Most of the research work appearing in this thesis has either already been published or is in several stages of publication at various venues. The relationship between these chapters and the publica-

Table 1.1: Fundamental Limits of Distributed Inference with Byzantine Data

Architecture	**Degrees of Freedom**	**Blinding Condition**	**Critical Power**
Parallel	Number of nodes B out of N	$\frac{B}{N} \geq \frac{1}{P_{1,0}+P_{0,1}}$ ***(Unique)***	$1/2$
Tree with unlabeled data	Set $\{B_k\}_{k=1}^K$ out of $\{N_k\}_{k=1}^K$	$\sum_{k=1}^K \left(\frac{B_k}{N_k}\sum_{i=k}^K N_i\right) \geq \frac{N}{2}$ ***(Non-unique)***	$\frac{N_1}{2\sum_{k=1}^K N_k}$
Tree with labeled data	Set $\{B_k\}_{k=1}^K$ out of $\{N_k\}_{k=1}^K$	$\sum_{j=1}^k \frac{B_j}{N_j} \geq \frac{1}{2},\ \forall k$ ***(Unique)***	$\frac{N_1}{2\sum_{k=1}^K N_k}$
Peer to Peer	Number of nodes B out of N	$\frac{B}{N} \geq \frac{\eta\sigma^2}{2\Delta}$ ***(Non-unique)***	$1/N$

Table 1.2: Proposed Byzantine Mitigation Schemes

Architecture	**System Component Improvement**	**Error Correcting Codes**	**System Redesign**
Parallel	Optimal fusion rule	Special case of tree with depth 1	Joint optimization of fusion rule and sensor thresholds
Tree	Optimal fusion rule Reputation based identification scheme	Distributed fusion using ECC	Robust topology design
Peer to Peer	Adaptive fusion rule Learning based identification scheme	Not considered	Robust consensus protocol for decentralized fusion

tions are as follows:

- **Chapter 3:** J6, J8, C4, C3
- **Chapter 4:** J9, C5
- **Chapter 5:** J5
- **Chapter 6:** J7
- **Chapter 7:** J2, C2
- **Chapter 8:** J1, J3, J4, C1, C6

JOURNAL PAPERS

J1 **B. Kailkhura**, Thakshila Wimalajeewa, and P. K. Varshney, "Collaborative Compressive Detection with Physical Layer Secrecy Constraints," IEEE Trans. Sig. Process., under review.

J2 **B. Kailkhura**, S. Brahma, and P. K. Varshney, "Consensus based Detection in the Presence of Data Falsification Attacks," to appear in IEEE Trans. Sig. Process.

J3 **B. Kailkhura**, S. Liu, Thakshila Wimalajeewa, and P. K. Varshney, "Measurement Matrix Design for Compressive Detection with Secrecy Guarantees,", to appear in IEEE Wireless Commun. Lett.

J4 **B. Kailkhura**, V. Sriram Siddhardh (Sid) Nadendla, and P. K. Varshney, "Distributed Inference in the Presence of Eavesdroppers: A Survey," IEEE Commun. Mag., vol. 53, no. 6, pp. 40 - 46, June, 2015.

J5 **B. Kailkhura**, S. Brahma, B. Dulek, Y. S. Han, and P. K. Varshney, "Distributed Detection in Tree-based Topologies: Byzantines and Mitigation Techniques," IEEE Trans. Inf. Forensics Security., vol. 10, no. 7, pp. 1499 - 1512, July, 2015.

J6 **B. Kailkhura**, Y. S. Han, S. Brahma, and P. K. Varshney, "Distributed Bayesian Detection with Byzantine Data," IEEE Trans. Signal Process., vol. 63, no. 19, pp. 5250 5263, Oct 1, 2015.

J7 **B. Kailkhura**, A. Vempaty, and P. K. Varshney, "Distributed Inference in Tree Networks using Coding Theory," IEEE Trans. Signal Process., vol. 63, no. 14, pp. 3715 - 3726, July 15, 2015.

J8 **B. Kailkhura**, Y. S. Han, S. Brahma, and P. K. Varshney, "Asymptotic Analysis of Distributed Bayesian Detection with Byzantine Data," IEEE Signal Process. Lett., vol. 22, no. 5, pp. 608-612, May 2015.

J9 **B. Kailkhura**, S. Brahma, Y. S. Han, and P. K. Varshney, "Distributed Detection in Tree Topologies with Byzantines," IEEE Trans. Signal Process., vol. 62, no. 12, pp. 3208-3219, June 15, 2014.

CONFERENCE PAPERS

C1 **B. Kailkhura**, L. Shen, T. Wimalajeewa, and P. K. Varshney, "Distributed Compressive Detection with Perfect Secrecy", IEEE 11th International Conference on Mobile Ad Hoc and Sensor Systems, Philadelphia, PA, 2014, pp. 674-679.

C2 **B. Kailkhura**, S. Brahma, and P. K. Varshney, "On Performance Analysis of Data Fusion schemes with Byzantines," IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), Florence, 2014, pp. 7411-7415.

C3 **B. Kailkhura**, Y. S. Han, S. Brahma, and P. K. Varshney, "On Covert Data Falsification Attacks on Distributed Detection Systems," 13th International Symposium on Communications and Information Technologies (ISCIT), Surat Thani, 2013, pp. 412-417.

C4 **B. Kailkhura**, S. Brahma, Y. S. Han, and P. K. Varshney, "Optimal Distributed Detection in the Presence of Byzantines," IEEE International Conference on Acoustics, Speech and Signal Processing, Vancouver, BC, 2013, pp. 2925-2929.

C5 **B. Kailkhura**, S. Brahma, and P. K. Varshney, "Optimal Byzantine Attacks on Distributed Detection in Tree-based Topologies," Computing, Networking and Communications (ICNC), 2013 International Conference on, San Diego, CA, 2013, pp. 227-231.

INVITED CONFERENCE PAPERS

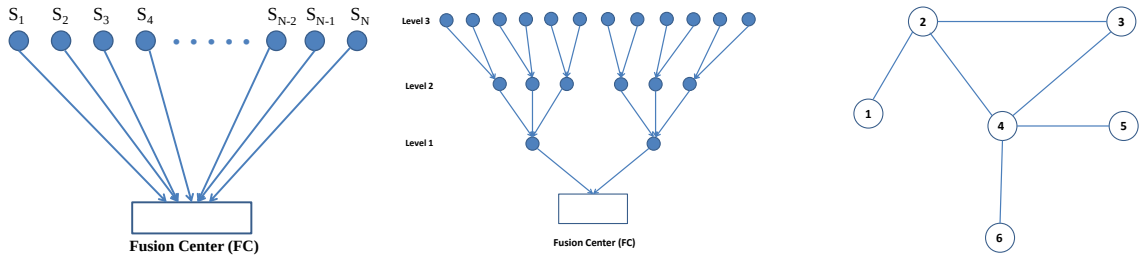
C6 **B. Kailkhura**, T. Wimalajeewa, and P. K. Varshney, "On Physical Layer Secrecy of Collaborative Compressive Detection," 48th Asilomar Conference on Signals, Systems and Computers, Pacific Grove, CA, 2014, pp. 51-55.

CHAPTER 2
BACKGROUND

As discussed in the previous chapter, distributed inference has been extensively studied by various authors over the past few decades. In the context of distributed inference with multiple sensors in a sensor network, a good survey can be found in [105], and references therein. However, limited work has focused on the case when these nodes are potentially unreliable and provide corrupt data. In this chapter, we present a quick background required for this thesis. In Sec. 2.1, we describe the general system model of the problems addressed in this thesis followed by taxonomy in Sec. 2.2. Some asymptotic performance metrics for distributed detection used in this thesis are presented in Sec. 2.3. Some background material is presented in Sec. 2.4 which is helpful in understanding the schemes proposed in the later chapters.

2.1 General Architecture

The generalized system model followed in this thesis comprises of a group of networked nodes which acquire observations regarding a phenomenon of interest (POI), collaborate with other nodes by sharing their local inference via different network topologies to make a global inference. These nodes can be honest or unreliable. For example, there may be some nodes that are providing false information. Some nodes can be genuinely interested in providing the right information



(a) (b) (c)

Fig. 2.1: (a) Parallel topology. (b) Tree topology. (c) Decentralized topology.

but due to erroneous sensing provide some unreliable information. Similarly, some nodes could be imperfect, or even faulty (e.g., stuck-at faults). Nodes are also prone to malicious attacks and could, therefore, be attacked by an external adversary and re-programmed to send flipped or altered versions of information. Note that, DINs can be organized in different topologies depending on the arrangement of nodes and existence of the FC. In most of the practical scenarios, DINs employ the following topologies

2.1.1 Network Topologies

Parallel Topology

Parallel topology, as depicted in Fig. 2.1(a), comprises of N nodes and a centralized FC. As described above, nodes collect the information simultaneously, carry out local processing and transmit the processed data directly to the FC where the final inference is made regarding the POI. Note that, nodes carry out local computation/processing independently without collaborating with each other.

Tree or Multi-hop Topology

Tree topology with depth K (greater than 1) comprises of N_k nodes at level $k \in \{1, \cdots, K\}$, and a FC at the root (or level 0) of the tree (see Fig. 2.1(b)). Observations are acquired and processed by leaf nodes at Level K and sent to their parent nodes, each of which may fuse all the messages it

receives with its own measurement (if any) and then forwards the new message to its parent node at the next level. This process takes place throughout the tree, culminating at the root (or FC) where the global inference is made. In this way, information from each node is aggregated at the FC sent via multi-hop paths.

Decentralized or Peer to Peer Topology

In a decentralized (or peer to peer topology), a centralized FC is not available. Network topology in such scenarios is modeled as a directed graph $G(V, E)$ with $|V| = N$ nodes (see Fig. 2.1(c)). The set of communication links in the network correspond to the set of edges E, where an edge exists if and only if there is a communication link between nodes to directly communicate with each other. In order to reach a global inference, peer-to-peer local information exchange schemes (e.g., consensus, gossip algorithm and diffusion) are employed where each node communicates only with its neighbors according to a pre-specified local fusion rule.

2.2 Taxonomy

We discuss some preliminaries for the distributed detection problem which is the major focus of this thesis.

2.2.1 Distributed Detection

Consider two hypotheses H_0 (signal is absent) and H_1 (signal is present). Also, consider a network comprised of a FC and a set of N nodes, which faces the task of determining which of the two hypotheses is true. The nodes observe the phenomenon, carry out local computations to decide the presence or absence of the phenomenon, and then send their local decisions to the FC that yields a final decision after processing the local decisions. Observations at the nodes are assumed to be conditionally independent and identically distributed given the hypothesis. We consider the communication channels to be error-free. Next, we describe the modus operandi of the nodes and

the FC in detail.

2.2.2 Modus Operandi of the Nodes

In this thesis, we consider that observations at the nodes are independent and identically distributed conditioned on the hypothesis. Based on the observations, each node i makes a one-bit local decision $v_i \in \{0, 1\}$ regarding the absence or presence of the phenomenon using the likelihood ratio test

$$\frac{p_{Yi}^{(1)}(y_i)}{p_{Yi}^{(0)}(y_i)} \underset{v_i=0}{\overset{v_i=1}{\gtrless}} \lambda, \tag{2.1}$$

where λ is the identical threshold[1] used at all the sensors and $p_{Yi}^{(k)}(y_i)$ is the conditional probability density function (PDF) of observation y_i under the hypothesis H_k. Each node i, after making its one-bit local decision v_i, sends $u_i \in \{0, 1\}$ to the FC, where $u_i = v_i$ if i is a reliable node, but for an unreliable node i, u_i need not be equal to v_i. We denote the probabilities of detection and false alarm of each node i in the network by $P_d = P(v_i = 1|H_1)$ and $P_f = P(v_i = 1|H_0)$, respectively, which hold for both reliable nodes as well as unreliable nodes.

2.2.3 Data Corruption model

In this thesis, we consider a probabilistic model for the data corrupting Byzantines. If a node is reliable(honest), then it transmits its own decision without altering it. However, a Byzantine node, in order to undermine the network performance, may alter its decision prior to transmission. In this thesis, we assume that each Byzantine decides to attack independently relying on its own observation and decision regarding the presence of the phenomenon. We define the following strategies $P_{j,1}^H$, $P_{j,0}^H$ and $P_{j,1}^B$, $P_{j,0}^B$ ($j \in \{0, 1\}$) for the honest and Byzantine nodes, respectively:
Honest nodes:

$$P_{1,1}^H = 1 - P_{0,1}^H = P^H(x = 1|y = 1) = 1 \tag{2.2}$$

$$P_{1,0}^H = 1 - P_{0,0}^H = P^H(x = 1|y = 0) = 0 \tag{2.3}$$

[1]It has been shown that the use of identical thresholds is asymptotically optimal [14].

Byzantine nodes:

$$P_{1,1}^{B} = 1 - P_{0,1}^{B} = P^{B}(x = 1|y = 1) \tag{2.4}$$

$$P_{1,0}^{B} = 1 - P_{0,0}^{B} = P^{B}(x = 1|y = 0) \tag{2.5}$$

where $P^{H}(x = a|y = b)$ ($P^{B}(x = a|y = b)$) is the probability that an honest (Byzantine) node sends a to the FC when its actual local decision is b.

2.2.4 Binary Hypothesis Testing at the Fusion Center

There are different hypothesis testing methods adopting various design rules in the literature. Here, we discuss two widely adopted hypothesis testing methods, i.e., the Bayesian test and the Neyman-Pearson test.

Bayesian Test

Here we focus on a Bayesian detection problem where the performance criterion at the FC is the probability of error. The FC receives decision vector, $\mathbf{u} = [u_1, \cdots, u_N]$, from the nodes and makes the global decision about the phenomenon by considering the maximum a *posteriori* probability (MAP) rule which is given by

$$P(H_1|\mathbf{u}) \underset{H_0}{\overset{H_1}{\gtrless}} P(H_0|\mathbf{u})$$

or equivalently,

$$\frac{P(\mathbf{u}|H_1)}{P(\mathbf{u}|H_0)} \underset{H_0}{\overset{H_1}{\gtrless}} \frac{P_0}{P_1}.$$

Since the u_i's are independent of each other, the MAP rule simplifies to a K-out-of-N fusion rule.

Neyman-Pearson Test

In many practical situations, the prior probabilities are unknown or difficult to estimate. In this case, the Neyman- Pearson (NP) test is introduced to maximize the probability of detection, while maintaining the probability of false alarm to be lower than a certain acceptable value. With simple

mathematical derivations, the resulting test is performed as follows

$$\frac{P(\mathbf{u}|H_1)}{P(\mathbf{u}|H_0)} \underset{H_0}{\overset{H_1}{\gtrless}} \lambda$$

where λ is the detection threshold calculated based on the maximum acceptable probability of false alarm.

2.3 Asymptotic Performance Metrics

In several cases, detection performance at the FC cannot be analyzed easily for the non-asymptotic regime. To gain insights into the performance, one can consider the asymptotic regime and employ the error exponents to be the network performance metric that characterizes detection performance.

2.3.1 Neyman-Pearson Case: Kullback-Leibler divergence (KLD)

Using Stein's lemma [23], we know that the Kullback-Leibler divergence (KLD) represents the best error exponent of the missed detection error probability in the NP setup. For a fixed false alarm probability, $P_F \leq \delta$, the missed detection probability for an optimal NP detector asymptotically behaves as

$$\lim_{N\to\infty} \frac{1}{N} \log P_M = -D(H_0\|H_1)$$

where N is the number of samples used for detection and $D(H_0\|H_1) = \sum_{j\in\{0,1\}} \pi_{j,0}^k \log \frac{\pi_{j,0}^k}{\pi_{j,1}^k}$ is the Kullback-Leibler divergence (KLD) and π_{j0} and π_{j1} are the conditional probabilities of $u_i = j$ given H_0 and H_1, respectively. A direct consequence of this statement is that P_M decays, as N grows to infinity, exponentially, i.e.,

$$P_M \approx f(N)e^{-D(H_0\|H_1)},$$

where $f(N)$ is a slow-varying function compared to the exponential, such that $\lim_{N\to\infty} \frac{1}{N} \log f(N) = 0$. Therefore, given a number of observations, the detection performance depends exclusively on the KLD between the hypotheses. We can conclude that the larger the KLD is, the less is the likelihood of mistaking H_0 with H_1 and, therefore, KLD can be used as a surrogate for the probability of missed detection during system design for a large network.[2]

2.3.2 Bayesian Case: Chernoff Information

Similarly, the Chernoff information represents the best error exponent of the error probability in the Bayesian setup. Formally, if $\mathbf{u}$ is a random vector having N statistically independent and identically distributed components, u_is, under both hypotheses, the optimal detector results in error probability that obeys the asymptotics

$$\lim_{N\to\infty} \frac{\ln P_E}{N} = -C(\pi_{1,1}, \pi_{1,0}), \tag{2.6}$$

where the Chernoff information C is defined as

$$C = \max_{0\leq t\leq 1} -\ln(\sum_{j\in\{0,1\}} \pi_{j0}^{t} \pi_{j1}^{1-t}). \tag{2.7}$$

π_{j0} and π_{j1} are the conditional probabilities of $u_i = j$ given H_0 and H_1, respectively.

2.4 Background Material: Distributed Classification Fusion using Error-Correcting Codes (DCFECC)

In [117], the authors proposed the DCFECC scheme for M-ary distributed classification using binary quantized local data for a parallel topology network. The idea behind the DCFECC scheme is to select a binary code matrix C to determine the local decision rules at the nodes, and to

[2]Kullback-Leibler divergence based detection approaches perform reasonably well even for a small size network as observed in [31, 66, 88, 98] .

perform fault-tolerant fusion at the FC. For a network with N nodes trying to distinguish among M hypotheses, the code matrix C is an $M \times N$ binary matrix. Each row of C corresponds to one of the M possible hypotheses $H_1, \cdots, H_M$ and each column represents the binary decision rule of the corresponding node. Given this code matrix, the node j sends its binary decision $u_j \in \{0, 1\}$ to the FC. After receiving the binary decisions $\mathbf{u} = (u_1, \cdots, u_N)$ from local nodes, the final classification decision is made at the FC using minimum Hamming distance based fusion given by:

Decide H_m where

$$m = \underset{1 \le l \le M}{\arg\min}\, d_H(\mathbf{u}, \mathbf{r}_l), \tag{2.8}$$

where $d_H(\mathbf{x}, \mathbf{y})$ is the Hamming distance between $\mathbf{x}$ and $\mathbf{y}$, and $\mathbf{r}_l = (c_{l1}, \cdots, c_{lN})$ is the lth row of C which corresponds to hypothesis H_l. The tie-break rule is to randomly pick a row of the code matrix C from those with the smallest Hamming distance to the received vector $\mathbf{u}$. The performance of the scheme depends on the code matrix C since it is used for designing the local decision rules as well as for the fusion rule at the FC. Several approaches to design the matrix C, e.g., based on simulated annealing and cyclic column replacement, were presented in [117].

For example, consider the code matrix used by a parallel network of $N = 7$ nodes performing an $(M = 4)$-ary classification problem

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

When the true hypothesis is H_1 corresponding to the first row, all the nodes are supposed to send the first element of their column. However, due to imperfect observations at the nodes, consider the case when the FC receives the vector $[1110101]$. The FC evaluates the Hamming distance between this received vector and each of the rows resulting in the Hamming distance values $(2, 4, 5, 3)$. Therefore, it decides the hypothesis corresponding to the first row, H_1, as the true hypothesis.

CHAPTER 3
DISTRIBUTED BAYESIAN DETECTION WITH CORRUPTED DATA: PARALLEL TOPOLOGY

3.1 Introduction

In this chapter, we consider the problem of distributed Bayesian detection in the presence of Byzantines in the parallel network. It is assumed that a fraction of the nodes in the network are compromised and reprogrammed by an adversary to transmit false information to the fusion center (FC) to degrade detection performance. The problem of distributed detection is formulated as a binary hypothesis test at the FC based on 1-bit data sent by the sensors. The expression for minimum attacking power required by the Byzantines to blind the FC is obtained. More specifically, we show that above a certain fraction of Byzantine attackers in the network, the detection scheme becomes completely incapable of utilizing the sensor data for detection. First, by employing Chernoff information as our performance metric, we analyze the problem and derive results for the asymptotic case. In practice, the FC and the Byzantines will optimize their utility by choosing their actions based on the knowledge of their opponent's behavior. This motivates us to address the question:

Table 3.1: Different scenarios based on the knowledge of the opponent's strategies

Cases	Attacker has the knowledge of the FC's strategies	FC has the knowledge of Attacker's strategies
Case 1	No	No
Case 2	Yes	No
Case 3	Yes	Yes
Case 4	No	Yes

what are the optimal attacking/defense strategies given the knowledge of the opponent's strategies? Study of these practically motivated questions requires non asymptotic analysis, which is systematically studied in this chapter. By assuming the error probability to be our performance metric, we analyze the problem in the non asymptotic regime. Observe that, the probability of error is a function of the fusion rule, which is under the control of the FC. This gives us an additional degree of freedom to analyze the Byzantine attack under different practical scenarios where the FC and the Byzantines may or may not have knowledge of their opponent's strategies. (For a description of different scenarios see Table 3.1). It is found that asymptotics-based results do not hold under several non-asymptotic scenarios. More specifically, when the FC does not have knowledge of attacker's strategies, results for the non-asymptotic case are different from those for the asymptotic case. However, if the FC has complete knowledge of the attacker's strategies and uses the optimal fusion rule to make the global decision, results obtained for this case are the same as those for the asymptotic case. Knowledge of the behavior of the attacker in the non-asymptotic regime enables the analysis of many related questions, such as the design of the optimal detector (fusion rule) and effects of strategic interaction between the FC and the attacker.

The rest of the chapter is organized as follows. Section 3.2.1 introduces our system model, including the Byzantine attack model. In Section 3.3, we provide the closed form expression for the critical power above which the FC becomes blind. In Section 3.4, we conduct the asymptotic analysis of the distributed Bayesian detection with Byzantine data. Next, we discuss our results based on non-asymptotic analysis of the distributed Bayesian detection system with Byzantine

data for different scenarios. In Section 3.5, we analyze the problem when Byzantines do not have any knowledge about the fusion rule used at the FC. Section 3.6 discusses the scenario where Byzantines have the knowledge about the fusion rule used at the FC, but the FC does not know the attacker's strategies. Next in Section 3.7, we extend our analysis to the scenario where both the FC and the attacker have the knowledge of their opponent's strategies and act strategically to optimize their utilities. Finally, Section 3.9 concludes the chapter.

3.2 Preliminaries

3.2.1 System Model

Consider two hypotheses H_0 (signal is absent) and H_1 (signal is present). Also, consider a parallel network (see Figure 2.1(a)), comprised of a central entity (known as the Fusion Center (FC)) and a set of N sensors (nodes), which faces the task of determining which of the two hypotheses is true. Prior probabilities of the two hypotheses H_0 and H_1 are denoted by P_0 and P_1, respectively. The sensors observe the phenomenon, carry out local computations to decide the presence or absence of the phenomenon, and then send their local decisions to the FC that yields a final decision after processing the local decisions. Observations at the nodes are assumed to be conditionally independent and identically distributed. A Byzantine attack on such a system compromises some of the nodes which may then intentionally send falsified local decisions to the FC to make the final decision incorrect. We assume that a fraction α of the N nodes which observe the phenomenon have been compromised by an attacker. Based on the observations, each node i makes a one-bit local decision $v_i \in \{0, 1\}$ regarding the absence or presence of the phenomenon using the likelihood ratio test as given in (2.1). Each node i, after making its one-bit local decision v_i, sends u_i to the FC, where $u_i = v_i$ if i is an uncompromised (honest) node, but for a compromised (Byzantine) node i, u_i need not be equal to v_i. we assume that each Byzantine decides to attack independently relying on its own observation and decision regarding the presence of the phenomenon. Specifically, we employ the Byzantine data model as given in Sec. 2.2.3. The strategies are denoted by

$P_{j,1}^H$, $P_{j,0}^H$ and $P_{j,1}^B$, $P_{j,0}^B$ ($j \in \{0,1\}$) for the honest and Byzantine nodes, respectively. The FC receives decision vector, $\mathbf{u} = [u_1, \cdots, u_N]$, from the nodes and makes the global decision about the phenomenon by considering the maximum a posterior probability (MAP) rule. Since the u_is are independent of each other, the MAP rule simplifies to a K-out-of-N fusion rule [104]. The global false alarm probability Q_F and detection probability Q_D are then given by[1]

$$Q_F = \sum_{i=K}^{N} \begin{pmatrix} N \\ i \end{pmatrix} (\pi_{1,0})^i (1-\pi_{1,0})^{N-i} \tag{3.1}$$

and

$$Q_D = \sum_{i=K}^{N} \begin{pmatrix} N \\ i \end{pmatrix} (\pi_{1,1})^i (1-\pi_{1,1})^{N-i}, \tag{3.2}$$

where π_{j0} and π_{j1} are the conditional probabilities of $u_i = j$ given H_0 and H_1, respectively. Specifically, $\pi_{1,0}$ and $\pi_{1,1}$ can be calculated as

$$\pi_{1,0} = \alpha(P_{1,0}(1-P_f) + (1-P_{0,1})P_f) + (1-\alpha)P_f \tag{3.3}$$

and

$$\pi_{1,1} = \alpha(P_{1,0}(1-P_d) + (1-P_{0,1})P_d) + (1-\alpha)P_d, \tag{3.4}$$

where α is the fraction of Byzantine nodes.[2]

The local probability of error as seen by the FC is defined as

$$P_e = P_0\pi_{1,0} + P_1\left(1-\pi_{1,1}\right) \tag{3.5}$$

[1]These expressions are valid under the assumption that $\alpha < 0.5$. Later in Section 3.7, we will generalize our result for any arbitrary α.

[2]The proposed analysis can be easily extended to the noisy channel case. For example, let us consider the Binary Symmetric Channel with crossover probabilities given by $(\hat{P_{1,0}}, \hat{P_{0,1}})$. Now, the conditional probability $\pi_{1,1}$ as given in (3.4) changes to:

$$\begin{aligned} \pi_{1,1} \;=\; & \alpha(1-\hat{P_{0,1}})[(1-P_{0,1})P_d + P_{1,0}(1-P_d)] + \alpha\hat{P_{1,0}}[P_{0,1}P_d + (1-P_{1,0})(1-P_d)] \\ & +(1-\alpha)[(1-\hat{P_{0,1}})P_d + 1\hat{,}0(1-Pd)] \end{aligned}$$

and similarly the expression for $\pi_{1,0}$ can be obtained. Using these expressions, the proposed analysis can be extended to the noisy case (BSC).

and the system wide probability of error at the FC is given by

$$P_E = P_0 Q_F + P_1 (1 - Q_D) . \tag{3.6}$$

Notice that, the system wide probability of error P_E is a function of the parameter K, which is under the control of the FC, and the parameters $(\alpha, P_{j,0}, P_{j,1})$ are under the control of the attacker. The FC and the Byzantines may or may not have knowledge of their opponent's strategy. In this chapter, we will analyze the problem of detection with Byzantine data under several different scenarios in the following sections. First, we will determine the minimum fraction of Byzantines needed to blind the decision fusion scheme.

3.3 Critical Power to Blind the Fusion Center

In this section, we determine the minimum fraction of Byzantine nodes needed to make the FC "blind" and denote it by α_{blind}. We say that the FC is blind if an adversary can make the data that the FC receives from the sensors such that no information is conveyed. In other words, the optimal detector at the FC cannot perform better than simply making the decision based on priors.

Lemma 3.3.1. *In Bayesian distributed detection, the minimum fraction of Byzantines needed to make the FC blind is* $\alpha_{blind} = 0.5$.

Proof. In the Bayesian framework, we say that the FC is 'blind', if the received data $\mathbf{u}$ does not provide any information about the hypotheses to the FC. That is, the condition to make the FC blind can be stated as

$$P(H_i|\mathbf{u}) = P(H_i) \text{ for } i = 0, 1 \tag{3.7}$$

It can be seen that (3.7) is equivalent to

$$\begin{aligned} & P(H_i|\mathbf{u}) = P(H_i) \\ \Leftrightarrow \quad & \frac{P(H_i)P(\mathbf{u}|H_i)}{P(\mathbf{u})} = P(H_i) \\ \Leftrightarrow \quad & P(\mathbf{u}|H_i) = P(\mathbf{u}). \end{aligned}$$

Thus, the FC becomes blind if the probability of receiving a given vector $\mathbf{u}$ is independent of the hypothesis present. In such a scenario, the best that the FC can do is to make decisions solely based on the priors, resulting in the most degraded performance at the FC. Now, using the conditional i.i.d. assumption, under which observations at the nodes are conditionally independent and identically distributed, condition (3.7) to make the FC blind becomes $\pi_{1,1} = \pi_{1,0}$. This is true only when

$$\alpha[P_{1,0}(P_f - P_d) + (1 - P_{0,1})(P_d - P_f)] + (1 - \alpha)(P_d - P_f) = 0.$$

Hence, the FC becomes blind if

$$\alpha = \frac{1}{(P_{1,0} + P_{0,1})}. \tag{3.8}$$

α in (3.8) is minimized when $P_{1,0}$ and $P_{0,1}$ both take their largest values, i.e., $P_{1,0} = P_{0,1} = 1$. Hence, $\alpha_{blind} = 0.5$. □

Next, we investigate how the Byzantines can launch an attack optimally considering that the parameter (K) is under the control of the FC. The detection performance at the FC in the presence of the Byzantines, however, cannot be analyzed easily for the non-asymptotic case. To gain insights into the degree to which an adversary can cause performance degradation, we consider the asymptotic regime, so that error probabilities may be approximated using large deviation analysis.

3.4 Asymptotic Analysis of Optimal Byzantine Attack

In this section, we look at the asymptotic scenario where the number of nodes in the network is large. In this setting, the asymptotic performance is measured in terms of the Chernoff information [20]. Chernoff information $\mathbb{C}$ between two joint distributions of statistically independent, identically distributed random variables is the sum of the marginal Chernoff information C. Since we assume that the nodes' observations are independent, Chernoff information can be expressed as $\mathbb{C} = NC$. From now onwards, we only look at the marginal Chernoff information and refer to it as the Chernoff information, since minimization or maximization of $\mathbb{C}$ is equivalent to minimization or maximization of the marginal Chernoff informations C.

From the Byzantine attacker's point of view, our goal is to find $P_{1,0}$ and $P_{0,1}$ that minimize Chernoff information C for a given value of α. Observe that, when $\alpha \geq 0.5$, Chernoff information can be minimized by simply making posterior probabilities equal to prior probabilities (we discuss this in more detail later in the section). However, for $\alpha < 0.5$, a closed form expression for Chernoff information is needed to find $P_{1,0}$ and $P_{0,1}$ that minimize C. To obtain the closed form expression of Chernoff information, the solution of an optimization problem is required: $\underset{0\leq t\leq 1}{max} - \ln(\sum_{j\in\{0,1\}} \pi_{j0}^{t}\pi_{j1}^{1-t})$. Next, we find a closed form expression for the Chernoff information when $\alpha < 0.5$.

3.4.1 Closed Form Expression for the Chernoff Information when $\alpha < 0.5$

In this subsection, we derive a closed form expression for the Chernoff information, when $\alpha < 0.5$.[3] Observe that the problem of finding the optimal t^* is equivalent to

$$\min_{0\leq t\leq 1} \ln\Big(\sum_{j\in\{0,1\}} \pi_{j0}^{t}\pi_{j1}^{1-t}\Big) \tag{3.9}$$

[3] Similar results can be derived for $\alpha \geq 0.5$.

which is a constrained minimization problem. To find t^*, we first perform unconstrained minimization (no constraint on the value of t) and later show that the solution of the unconstrained optimization problem is the same as the solution of the constrained optimization problem. In other words, the optimal t^* is the same for both cases.

By observing that logarithm is an increasing function, the optimization problem as given in (3.9) is equivalent to

$$\min_t [\pi_{1,0}^t \pi_{1,1}^{1-t} + (1-\pi_{1,0})^t (1-\pi_{1,1})^{1-t}]. \tag{3.10}$$

Now, performing the first derivative, we have

$$\begin{aligned} & \frac{d}{dt}[\pi_{1,0}^t \pi_{1,1}^{1-t} + (1-\pi_{1,0})^t (1-\pi_{1,1})^{1-t}] \\ = \; & \pi_{1,1} \left(\frac{\pi_{1,0}}{\pi_{1,1}}\right)^t \ln\left(\frac{\pi_{1,0}}{\pi_{1,1}}\right) + (1-\pi_{1,1}) \left(\frac{1-\pi_{1,0}}{1-\pi_{1,1}}\right)^t \ln\left(\frac{1-\pi_{1,0}}{1-\pi_{1,1}}\right). \end{aligned} \tag{3.11}$$

The first derivative (3.11) is set to zero to find the critical points of the function:

$$\left(\frac{(1/\pi_{1,0})-1}{(1/\pi_{1,1})-1}\right)^t = \frac{\ln(\pi_{1,1}/\pi_{1,0})}{\ln((1-\pi_{1,0})/(1-\pi_{1,1}))} \left(\frac{\pi_{1,1}}{1-\pi_{1,1}}\right). \tag{3.12}$$

After some simplification, t^* which satisfies (3.12) turns out to be

$$t^* = \frac{\ln\left(\dfrac{\ln(\pi_{1,1}/\pi_{1,0})}{\ln((1-\pi_{1,0})/(1-\pi_{1,1}))} \dfrac{\pi_{1,1}}{1-\pi_{1,1}}\right)}{\ln\left(\dfrac{(1/\pi_{1,0})-1}{(1/\pi_{1,1})-1}\right)}. \tag{3.13}$$

To determine whether the critical point is a minimum or a maximum, we perform the second derivative test. Since

$$\begin{aligned} & \frac{d^2}{d^2 t}[\pi_{1,0}^t \pi_{1,1}^{1-t} + (1-\pi_{1,0})^t (1-\pi_{1,1})^{1-t}] \\ = \; & \pi_{1,1} \left(\frac{\pi_{1,0}}{\pi_{1,1}}\right)^t \left(\ln\frac{\pi_{1,0}}{\pi_{1,1}}\right)^2 + (1-\pi_{1,1}) \left(\frac{1-\pi_{1,0}}{1-\pi_{1,1}}\right)^t \left(\ln\frac{1-\pi_{1,0}}{1-\pi_{1,1}}\right)^2 \end{aligned} \tag{3.14}$$

is greater than zero, t^* as given in (3.13) minimizes (3.10). Since $0 \leq t^* \leq 1$ (See proof in Appendix A.1), t^* as given in (3.13) is also the solution of (3.9).

3.4.2 Minimization of Chernoff Information

First, we minimize Chernoff information for $\alpha < 0.5$. Later in the section, we generalize our results for any arbitrary α. We formally state the problem as

$$\begin{aligned} \underset{P_{1,0},P_{0,1}}{\text{minimize}} \quad & -\ln\Big(\sum_{j\in\{0,1\}} \pi_{j0}^{t^*}\pi_{j1}^{1-t^*}\Big) \\ \text{subject to} \quad & 0 \leq P_{1,0} \leq 1 \\ & 0 \leq P_{0,1} \leq 1 \end{aligned}$$

Since logarithm is an increasing function, Problem 3.4.2 is equivalent to the following problem:

$$\begin{aligned} \underset{P_{1,0},P_{0,1}}{\text{maximize}} \quad & \pi_{1,0}^{t^*}\pi_{1,1}^{1-t^*} + (1-\pi_{1,0})^{t^*}(1-\pi_{1,1})^{1-t^*} \\ \text{subject to} \quad & 0 \leq P_{1,0} \leq 1 \\ & 0 \leq P_{0,1} \leq 1 \end{aligned}$$

where $\alpha < 0.5$ and t^* is as given in (3.13).

Let us denote $\tilde{C} = \pi_{1,0}^{t^*}\pi_{1,1}^{1-t^*} + (1-\pi_{1,0})^{t^*}(1-\pi_{1,1})^{1-t^*}$. Observe that, maximization of $\tilde{C}$ is equivalent to the minimization of Chernoff information C. Next, in Lemma 3.4.1 we present the properties of Chernoff information C (for the case when $\alpha < 0.5$) with respect to $(P_{1,0}, P_{0,1})$ that enable us to find optimal attacking strategies in this case.

Lemma 3.4.1. *Let $\alpha < 0.5$ and assume that the optimal t^* is used in the expression for the Chernoff information. Then, the Chernoff information, C, is a monotonically decreasing function of $P_{1,0}$ for a fixed $P_{0,1}$. Conversely, the Chernoff information is also a monotonically decreasing function of $P_{0,1}$ for a fixed $P_{1,0}$.*

Proof. See Appendix A.2. □

Next, using Lemma 3.4.1, we present the optimal attacking strategies $P_{1,0}$ and $P_{0,1}$ that minimize the Chernoff information, C, for $0 \leq \alpha \leq 1$.

Theorem 3.4.2. *Optimal attacking strategies, $(P^*_{1,0}, P^*_{0,1})$, which minimize the Chernoff information are*

$$(P^*_{1,0}, P^*_{0,1}) \begin{cases} (p_{1,0}, p_{0,1}) & \text{if } \alpha \geq 0.5 \\ (1,1) & \text{if } \alpha < 0.5 \end{cases},$$

where, $(p_{1,0}, p_{0,1})$ satisfy $\alpha(p_{1,0} + p_{0,1}) = 1$.

Proof. The minimum value of C is zero and it occurs when $\pi_{1,1} = \pi_{1,0}$. By (3.3) and (3.4), $\pi_{1,1} = \pi_{1,0}$ implies

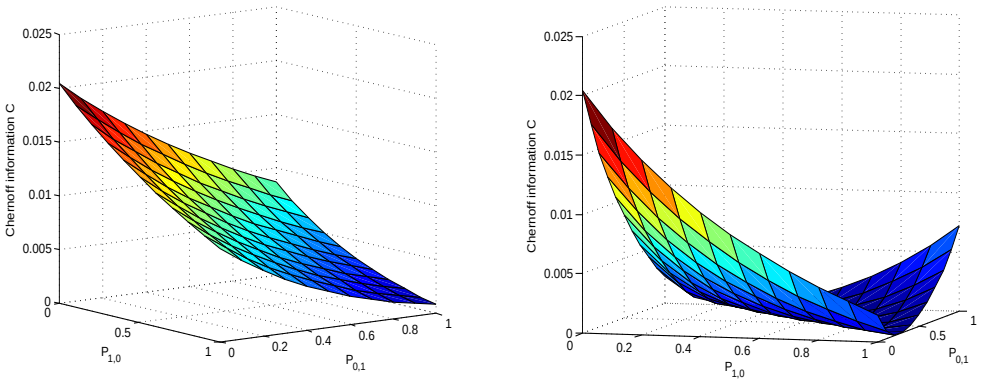
$$\alpha(P_{1,0} + P_{0,1}) = 1. \tag{3.15}$$

From (3.15), when $\alpha \geq 0.5$, the attacker can always find flipping probabilities that make the Chernoff information equal to zero. When $\alpha = 0.5$, $P_{1,0} = P_{0,1} = 1$ is the optimal strategy. When $\alpha > 0.5$, any pair which satisfies $P_{1,0} + P_{0,1} = \frac{1}{\alpha}$ is the optimal strategy. However, when $\alpha < 0.5$, (3.15) can not be satisfied or in other words Byzantines can not make $C = 0$ since $\pi_{1,1}$ can not be made equal to $\pi_{1,0}$. By Lemma 3.4.1, when $\alpha < 0.5$, the optimal attacking strategy, $(P_{1,0}, P_{0,1})$, that minimizes the Chernoff information is $(1, 1)$. □

Next, to gain insight into Theorem 3.4.2, we present illustrative examples that corroborate our results.

3.4.3 Illustrative Examples

In Figure 3.1(a), we plot the Chernoff information as a function of $(P_{1,0}, P_{0,1})$ for $(P_d = 0.6, P_f = 0.4)$ and $\alpha = 0.4$. It can be observed that for a fixed $P_{0,1}$ ($P_{1,0}$), the Chernoff information C is a monotonically decreasing function of $P_{1,0}$, $P_{0,1}$ (as has been shown in Lemma 3.4.1). In other words, when $\alpha = 0.4$, the attacking strategy, $(P_{1,0}, P_{0,1})$, that minimizes the Chernoff information C is $(1, 1)$.



(a) (b)

Fig. 3.1: (a) Chernoff information as a function of $(P_{1,0}, P_{0,1})$ for $\alpha = 0.4$. (b) Chernoff information as a function of $(P_{1,0}, P_{0,1})$ for $\alpha = 0.8$.

Similarly, in Figure 3.1(b), we consider the scenario when the fraction of Byzantines in the network is $\alpha = 0.8$. It can be seen from Figure 3.1(b) that the minimum value of the Chernoff information in this case is $C = 0$. Notice that, the attacking strategy, $(P_{1,0}, P_{0,1})$ that makes $C = 0$ is not unique in this case. It can be verified that any attacking strategy which satisfies $P_{1,0} + P_{0,1} = \frac{1}{0.8}$ would make $C = 0$. Thus, results presented in Figures 3.1(a) and 3.1(b) corroborate our theoretical result presented in Theorem 3.4.2.

Next, we investigate how the Byzantines can launch an attack optimally considering that the parameter (K) is under the control of the FC. By assuming error probability to be our performance metric, we analyze the non-asymptotic regime. Observe that the probability of error is dependent on the fusion rule. This gives us an additional degree of freedom to analyze the Byzantine attack under different scenarios where the FC and the Byzantines may or may not have knowledge of their opponent's strategies.

3.5 Optimal Attacking Strategies without the knowledge of Fusion Rule

In practice, the Byzantine attacker may not have the knowledge about the fusion rule, i.e., the value of K, used by the FC. In such scenarios, we obtain the optimal attacking strategy for Byzantines by maximizing the local probability of error as seen by the FC, which is independent of the fusion rule K. We formally state the problem as

$$\begin{aligned} \underset{P_{1,0},P_{0,1}}{\text{maximize}} \quad & P_0\pi_{1,0} + P_1(1-\pi_{1,1}) \\ \text{subject to} \quad & 0 \leq P_{1,0} \leq 1 \\ & 0 \leq P_{0,1} \leq 1 \end{aligned}$$

To solve the problem, we analyze the properties of the objective function, $P_e = P_0\pi_{1,0} + P_1(1-\pi_{1,1})$, with respect to $(P_{1,0}, P_{0,1})$. Notice that

$$\frac{dP_e}{P_{1,0}} = P_0\alpha(1-P_f) - P_1\alpha(1-P_d) \tag{3.16}$$

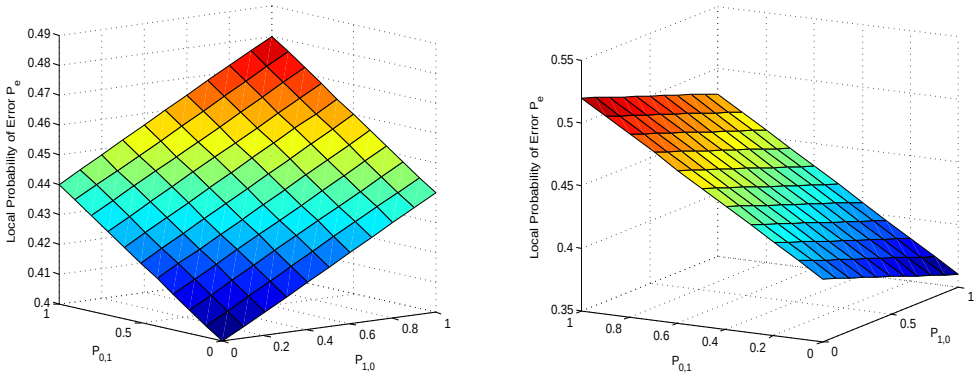
and

$$\frac{dP_e}{P_{0,1}} = -P_0\alpha P_f + P_1\alpha P_d. \tag{3.17}$$

By utilizing monotonicity properties of the objective function with respect to $P_{1,0}$ and $P_{0,1}$ ((3.16) and (3.17)), we present the solution of the Problem 3.5 in Table 3.2. Notice that, when $\frac{P_d}{P_f} < \frac{P_0}{P_1} < \frac{1-P_d}{1-P_f}$, both (3.16) and (3.17) are less than zero. P_e then becomes a strictly decreasing function of $P_{1,0}$ as well as $P_{0,1}$. Hence, to maximize P_e, the attacker needs to choose $(P_{1,0}, P_{0,1}) = (0,0)$. However, the condition $\frac{P_d}{P_f} < \frac{P_0}{P_1} < \frac{1-P_d}{1-P_f}$ holds iff $P_d < P_f$ and, therefore, is not admissible. Similar arguments lead to the rest of results given in Table 3.2. Note that, if there is an equality in the conditions mentioned in Table 3.2, then the solution will not be unique. For example, $\left(\frac{dP_e}{P_{0,1}} = 0\right) \Leftrightarrow \left(\frac{P_0}{P_1} = \frac{1-P_d}{1-P_f}\right)$ implies that the P_e is constant as a function of $P_{0,1}$. In other words, the attacker will be indifferent in choosing the parameter $P_{0,1}$ because any value of $P_{0,1}$ will

Table 3.2: Soultion Of Maximizing Local Error P_e Problem

$P_{1,0}$	$P_{0,1}$	Condition
0	0	$\frac{P_d}{P_f} < \frac{P_0}{P_1} < \frac{1-P_d}{1-P_f}$
0	1	$\frac{P_d}{P_f} > \frac{P_0}{P_1} < \frac{1-P_d}{1-P_f}$
1	0	$\frac{P_d}{P_f} < \frac{P_0}{P_1} > \frac{1-P_d}{1-P_f}$
1	1	$\frac{P_d}{P_f} > \frac{P_0}{P_1} > \frac{1-P_d}{1-P_f}$



(a) (b)

Fig. 3.2: (a) P_e as a function of $(P_{1,0}, P_{0,1})$ when $P_0 = P_1 = 0.5$. (b) P_e as a function of $(P_{1,0}, P_{0,1})$ when $P_0 = 0.1, P_1 = 0.9$.

result in the same probability of error.

Next, to gain insight into the solution, we present illustrative examples that corroborate our results.

3.5.1 Illustrative Examples

In Figure 3.2(a), we plot the local probability of error P_e as a function of $(P_{1,0}, P_{0,1})$ when $(P_0 = P_1 = 0.5)$. We assume that the local probability of detection is $P_d = 0.8$ and the local probability of false alarm is $P_f = 0.1$ such that $\frac{P_d}{P_f} = 8$, $\frac{1-P_d}{1-P_f} = .2222$, and $\frac{P_0}{P_1} = 1$. Clearly, $\frac{P_d}{P_f} > \frac{P_0}{P_1} > \frac{1-P_d}{1-P_f}$ and it implies that the optimal attacking strategy is $(P_{1,0}, P_{0,1}) = (1, 1)$, which can be verified from

Figure 3.2(a).

In Figure 3.2(b), we study the local probability of error P_e as a function of the attacking strategy $(P_{1,0}, P_{0,1})$ when $(P_0 = 0.1, P_1 = 0.9)$. We assume that the local probability of detection is $P_d = 0.8$ and the local probability of false alarm is $P_f = 0.1$ such that $\frac{P_d}{P_f} = 8$, $\frac{1-P_d}{1-P_f} = .2222$, and $\frac{P_0}{P_1} = .1111$. Clearly, $\frac{P_d}{P_f} > \frac{P_0}{P_1} < \frac{1-P_d}{1-P_f}$ implies that the optimal attacking strategy is $(P_{1,0}, P_{0,1}) = (0, 1)$, which can be verified from Figure 3.2(b). These results corroborate our theoretical results presented in Table 3.2.

In the next section, we investigate the scenario where Byzantines are aware of the fusion rule K used at the FC and can use this knowledge to provide false information in an optimal manner to blind the FC. However, the FC does not have the knowledge of Byzantine's attacking strategies $(\alpha, P_{j,0}, P_{j,1})$ and does not optimize against Byzantine's behavior. Since majority rule is a widely used fusion rule [46, 92, 125], we assume that the FC uses the majority rule to make the global decision.

3.6 Optimal Byzantine Attacking Strategies with Knowledge of Majority Fusion Rule

In this section, we investigate optimal Byzantine attacking strategies in a distributed detection system, with the attacker having knowledge about the fusion rule used at the FC. However, we assume that the FC is not strategic in nature, and uses a majority rule, without trying to optimize against the Byzantine's behavior. We consider both the FC and the Byzantine to be strategic in Section 3.7. The performance criterion at the FC is assumed to be the probability of error P_E.

For a fixed fusion rule (K^*), which, as mentioned before, is assumed to be the majority rule $K^* = \lceil \frac{N+1}{2} \rceil$, P_E varies with the parameters $(\alpha, P_{j,0}, P_{j,1})$ which are under the control of the

attacker. The Byzantine attack problem can be formally stated as follows:

$$\begin{aligned} \underset{P_{j,0},P_{j,1}}{\text{maximize}} \quad & P_E(\alpha, P_{j,0}, P_{j,1}) \\ \text{subject to} \quad & 0 \leq P_{j,0} \leq 1 \\ & 0 \leq P_{j,1} \leq 1. \end{aligned} \tag{3.18}$$

For a fixed fraction of Byzantines α, the attacker wants to maximize the probability of error P_E by choosing its attacking strategy $(P_{j,0}, P_{j,1})$ optimally. We assume that the attacker is aware of the fact that the FC is using the majority rule for making the global decision. Before presenting our main results for Problem 3.18, we make an assumption that will be used in the theorem.

Assumption 3.6.1. *We assume that* $\alpha < \min\{(0.5 - P_f), (1 - (m/P_d))\}$,[4] *where* $m = \frac{N}{2N-2}$.

A consequence of this assumption is $\pi_{1,1} > m$, which can be shown as follows. By (3.4), we have

$$\begin{aligned} \pi_{1,1} &= \alpha(P_{1,0}(1 - P_d) + (1 - P_{0,1})P_d) + (1 - \alpha)P_d \\ &= \alpha P_{1,0}(1 - P_d) - \alpha P_d P_{0,1} + P_d \\ &\geq -\alpha P_d P_{0,1} + P_d \geq P_d(1 - \alpha) > m. \end{aligned} \tag{3.19}$$

Eq. (3.19) is true because $\alpha < \min\{(0.5 - P_f), (1 - (m/P_d))\} \leq (1 - (m/P_d))$. Another consequence of this assumption is $\pi_{1,0} < 0.5$, which can be shown as follows. From (3.3), we have

$$\begin{aligned} \pi_{1,0} &= \alpha(P_{1,0}(1 - P_f) + (1 - P_{0,1})P_f) + (1 - \alpha)P_f \\ &= \alpha P_{1,0} - \alpha P_f(P_{1,0} + P_{0,1}) + P_f \\ &\leq \alpha + P_f < 0.5. \end{aligned} \tag{3.20}$$

[4]Condition $\alpha < \min\{(0.5 - P_f), (1 - (m/P_d))\}$, where $m = \frac{N}{2N-2} > 0.5$, suggests that as N tends to infinity, $m = \dfrac{N}{2N-2}$ tends to 0.5. When P_d tends to 1 and P_f tends to 0, the above condition becomes $\alpha < 0.5$.

Eq. (3.20) is true because $\alpha < \min\{(0.5 - P_f), (1 - (m/P_d))\} \leq (0.5 - P_f)$.

Next, we analyze the properties of P_E with respect to $(P_{1,0}, P_{0,1})$ under our assumption that enables us to find the optimal attacking strategies.

Lemma 3.6.2. *Assume that the FC employs the majority fusion rule K^* and $\alpha < \min\{(0.5 - P_f), (1 - (m/P_d))\}$, where $m = \frac{N}{2N-2}$. Then, for any fixed value of $P_{0,1}$, the error probability P_E at the FC is a quasi-convex function of $P_{1,0}$.*

Proof. A function $f(P_{1,0})$ is quasi-convex if, for some $P^*_{1,0}$, $f(P_{1,0})$ is non-increasing for $P_{1,0} \leq P^*_{1,0}$ and $f(P_{1,0})$ is non-decreasing for $P_{1,0} \geq P^*_{1,0}$. In other words, the lemma is proved if $\frac{dP_E}{dP_{1,0}} \leq 0$ (or $\frac{dP_E}{dP_{1,0}} \geq 0$) for all $P_{1,0}$, or if for some $P^*_{1,0}$, $\frac{dP_E}{dP_{1,0}} \leq 0$ when $P_{1,0} \leq P^*_{1,0}$ and $\frac{dP_E}{dP_{1,0}} \geq 0$ when $P_{1,0} \geq P^*_{1,0}$. First, we calculate the partial derivative of P_E with respect to $P_{1,0}$ for an arbitrary K as follows:

$$\frac{dP_E}{dP_{1,0}} = P_0 \frac{dQ_F}{dP_{1,0}} - P_1 \frac{dQ_D}{dP_{1,0}}. \tag{3.21}$$

The detailed derivation of $\frac{dP_E}{dP_{1,0}}$ is given in Appendix A.5 and we present a summary of the main results below.

$$\frac{dQ_F}{dP_{1,0}} = \alpha(1 - P_f)N \begin{pmatrix} N-1 \\ K-1 \end{pmatrix} (\pi_{1,0})^{K-1} (1 - \pi_{1,0})^{N-K}, \tag{3.22}$$

$$\frac{dQ_D}{dP_{1,0}} = \alpha(1 - P_d)N \begin{pmatrix} N-1 \\ K-1 \end{pmatrix} (\pi_{1,1})^{K-1} (1 - \pi_{1,1})^{N-K}, \tag{3.23}$$

and

$$\begin{aligned} \frac{dP_E}{dP_{1,0}} = &-P_1\alpha(1 - P_d)N \begin{pmatrix} N-1 \\ K-1 \end{pmatrix} (\pi_{1,1})^{K-1} (1 - \pi_{1,1})^{N-K} \\ &+ P_0\alpha(1 - P_f)N \begin{pmatrix} N-1 \\ K-1 \end{pmatrix} (\pi_{1,0})^{K-1} (1 - \pi_{1,0})^{N-K}. \end{aligned} \tag{3.24}$$

$\frac{dP_E}{dP_{1,0}}$ given in (3.24) can be reformulated as follows:

$$\frac{dP_E}{dP_{1,0}} = g\left(P_{1,0}, K, \alpha\right)\left(e^{r(P_{1,0},K,\alpha)} - 1\right), \tag{3.25}$$

where

$$g\left(P_{1,0}, K, \alpha\right) = N \begin{pmatrix} N-1 \\ K-1 \end{pmatrix} P_1 \alpha (1-P_d)(\pi_{1,1})^{K-1}(1-\pi_{1,1})^{N-K} \tag{3.26}$$

and

$$\begin{aligned} r\left(P_{1,0}, K, \alpha\right) &= \ln\left(\frac{P_0}{P_1}\frac{1-P_f}{1-P_d}\left(\frac{\pi_{1,0}}{\pi_{1,1}}\right)^{(K-1)}\left(\frac{1-\pi_{1,0}}{1-\pi_{1,1}}\right)^{(N-K)}\right) \\ &= \ln\frac{P_0}{P_1}\frac{1-P_f}{1-P_d} + (K-1)\ln\frac{\pi_{1,0}}{\pi_{1,1}} + (N-K)\ln\frac{1-\pi_{1,0}}{1-\pi_{1,1}}. \end{aligned} \tag{3.27}$$

It can be seen that $g\left(P_{1,0}, K, \alpha\right) \geq 0$ so that the sign of $\frac{dP_E}{dP_{1,0}}$ depends only on the value of $r\left(P_{1,0}, K, \alpha\right)$. To prove that P_E is a quasi-convex function of $P_{1,0}$ when the majority rule K^* is used at the FC, it is sufficient to show that $r\left(P_{1,0}, K^*, \alpha\right)$ is a non-decreasing function. Differentiating $r\left(P_{1,0}, K^*, \alpha\right)$ with respect to $P_{1,0}$, we get

$$\begin{aligned} \frac{dr\left(P_{1,0}, K^*, \alpha\right)}{dP_{1,0}} &= (K^*-1)\left(\frac{\alpha(1-P_f)}{\pi_{1,0}} - \frac{\alpha(1-P_d)}{\pi_{1,1}}\right) + (N-K^*)\left(\frac{\alpha(1-P_d)}{1-\pi_{1,1}} - \frac{\alpha(1-P_f)}{1-\pi_{1,0}}\right) \\ &= (K^*-1)\alpha\left(\frac{1-P_f}{\pi_{1,0}} - \frac{1-P_d}{\pi_{1,1}}\right) - (N-K^*)\alpha\left(\frac{1-P_f}{1-\pi_{1,0}} - \frac{1-P_d}{1-\pi_{1,1}}\right). \end{aligned} \tag{3.28}$$

It can be shown that $\frac{dr\left(P_{1,0}, K^*, \alpha\right)}{dP_{1,0}} > 0$ (see Appendix A.4) and this completes the proof. □

Quasi-convexity of P_E over $P_{1,0}$ implies that the maximum of the function occurs on the corners, i.e., $P_{1,0} = 0$ or 1 (may not be unique). Next, we analyze the properties of P_E with respect to $P_{0,1}$.

Lemma 3.6.3. *Assume that the FC employs the majority fusion rule K^* and $\alpha < \min\{(0.5 - P_f), (1 - (m/P_d))\}$, where $m = \frac{N}{2N-2}$. Then, the probability of error P_E at the FC is a quasi-*

convex function of $P_{0,1}$ for a fixed $P_{1,0}$.

Proof. For a fixed $P_{1,0}$, we have

$$(\pi_{1,0})' = d\pi_{1,0}/dP_{0,1} = \alpha(-P_f). \tag{3.29}$$

By a similar argument as given in Appendix A.5, for an arbitrary K we have

$$\frac{dP_E}{dP_{0,1}} = P_1\alpha P_d N \begin{pmatrix} N-1 \\ K-1 \end{pmatrix} (\pi_{1,1})^{K-1} (1-\pi_{1,1})^{N-K}$$

$$- P_0\alpha P_f N \begin{pmatrix} N-1 \\ K-1 \end{pmatrix} (\pi_{1,0})^{K-1} (1-\pi_{1,0})^{N-K}. \tag{3.30}$$

$\frac{dP_E}{dP_{0,1}}$ given in (3.30) can be reformulated as follows:

$$\frac{dP_E}{dP_{0,1}} = g\left(P_{0,1}, K, \alpha\right)\left(e^{r(P_{0,1},K,\alpha)} - 1\right), \tag{3.31}$$

where

$$g\left(P_{0,1}, K, \alpha\right) = N \begin{pmatrix} N-1 \\ K-1 \end{pmatrix} P_0\alpha P_f (\pi_{1,0})^{K-1}(1-\pi_{1,0})^{N-K} \tag{3.32}$$

and

$$\begin{aligned} r\left(P_{0,1}, K, \alpha\right) &= \ln\left(\frac{P_1}{P_0}\frac{P_d}{P_f}\left(\frac{\pi_{1,1}}{\pi_{1,0}}\right)^{(K-1)}\left(\frac{1-\pi_{1,1}}{1-\pi_{1,0}}\right)^{(N-K)}\right) \\ &= \ln\frac{P_1}{P_0}\frac{P_d}{P_f} + (K-1)\ln\frac{\pi_{1,1}}{\pi_{1,0}} + (N-K)\ln\frac{1-\pi_{1,1}}{1-\pi_{1,0}}. \end{aligned} \tag{3.33}$$

It can be seen that $g\left(P_{0,1}, K, \alpha\right) \geq 0$ such that the sign of $\frac{dP_E}{dP_{0,1}}$ depends on the value of $r\left(P_{0,1}, K, \alpha\right)$. To prove that P_E is a quasi-convex function of $P_{1,0}$ when the majority rule K^* is used at the FC, it is sufficient to show that $r\left(P_{0,1}, K^*, \alpha\right)$ is a non-decreasing function. Differentiating $r\left(P_{0,1}, K^*, \alpha\right)$

with respect to $P_{0,1}$, we get

$$\frac{dr\left(P_{0,1},K^*,\alpha\right)}{dP_{0,1}} = (K^*-1)\left(\frac{\alpha P_f}{\pi_{1,0}} - \frac{\alpha P_d}{\pi_{1,1}}\right) + (N-K^*)\left(\frac{\alpha P_d}{1-\pi_{1,1}} - \frac{\alpha P_f}{1-\pi_{1,0}}\right) \tag{3.34}$$

$$= (N-K^*)\alpha\left(\frac{P_d}{1-\pi_{1,1}} - \frac{P_f}{1-\pi_{1,0}}\right) - (K^*-1)\alpha\left(\frac{P_d}{\pi_{1,1}} - \frac{P_f}{\pi_{1,0}}\right). \tag{3.35}$$

In the following, we show that

$$\frac{dr\left(P_{0,1},K^*,\alpha\right)}{dP_{0,1}} > 0, \tag{3.36}$$

i.e., $r\left(P_{0,1},K^*,\alpha\right)$ is non-decreasing. It is sufficient to show that

$$(N-K^*)\left(\frac{P_d}{1-\pi_{1,1}} - \frac{P_f}{1-\pi_{1,0}}\right) > (K^*-1)\left(\frac{P_d}{\pi_{1,1}} - \frac{P_f}{\pi_{1,0}}\right). \tag{3.37}$$

First, we consider the case when there are an even number of nodes in the network and majority fusion rule is given by $K^* = \dfrac{N}{2}+1$. Since $0 \le \pi_{1,0} < \pi_{1,1} \le 1$ and $N \ge 2$, we have

$$\begin{aligned}
&\left(1-\frac{2}{N}\right)\frac{\pi_{1,1}\pi_{1,0}}{(1-\pi_{1,1})(1-\pi_{1,0})} > -1 \\
\Leftrightarrow\ &\left(1-\frac{2}{N}\right)\left[\frac{1}{1-\pi_{1,1}} - \frac{1}{1-\pi_{1,0}}\right] > \left[\frac{1}{\pi_{1,1}} - \frac{1}{\pi_{1,0}}\right] \\
\Leftrightarrow\ &\left[\left(1-\frac{2}{N}\right)\frac{1}{1-\pi_{1,1}} - \frac{1}{\pi_{1,1}}\right] > \left[\left(1-\frac{2}{N}\right)\frac{1}{1-\pi_{1,0}} - \frac{1}{\pi_{1,0}}\right].
\end{aligned} \tag{3.38}$$

Using the fact that $\dfrac{P_d}{P_f} > 1$, $\pi_{1,1} > \frac{N}{2N-2}$, and $K^* = \dfrac{N}{2}+1$, (3.38) becomes

$$\begin{aligned}
&\frac{P_d}{P_f}\left[\left(1-\frac{2}{N}\right)\frac{1}{1-\pi_{1,1}} - \frac{1}{\pi_{1,1}}\right] > \left[\left(1-\frac{2}{N}\right)\frac{1}{1-\pi_{1,0}} - \frac{1}{\pi_{1,0}}\right] \\
\Leftrightarrow\ &\left(1-\frac{2}{N}\right)\frac{P_d}{1-\pi_{1,1}} - \frac{P_d}{\pi_{1,1}} > \left(1-\frac{2}{N}\right)\frac{P_f}{1-\pi_{1,0}} - \frac{P_f}{\pi_{1,0}} \\
\Leftrightarrow\ &(N-K^*)\left(\frac{P_d}{1-\pi_{1,1}} - \frac{P_f}{1-\pi_{1,0}}\right) > (K^*-1)\left(\frac{P_d}{\pi_{1,1}} - \frac{P_f}{\pi_{1,0}}\right).
\end{aligned} \tag{3.39}$$

Next, we consider the case when there are odd number of nodes in the network and majority

fusion rule is given by $K^* = \dfrac{N+1}{2}$. By using the fact that $\frac{\pi_{1,0}}{\pi_{1,1}} > \frac{P_f}{P_d}$, it can be seen that the right-hand side of (3.39) is nonnegative. Hence, from (3.39), we have

$$\begin{aligned}
&\left(\frac{N}{2}-1\right)\left(\frac{P_d}{1-\pi_{1,1}}-\frac{P_f}{1-\pi_{1,0}}\right) > \frac{N}{2}\left(\frac{P_d}{\pi_{1,1}}-\frac{P_f}{\pi_{1,0}}\right) \\
\Leftrightarrow\ &\left(\frac{N-1}{2}\right)\left(\frac{P_d}{1-\pi_{1,1}}-\frac{P_f}{1-\pi_{1,0}}\right) > \left(\frac{N-1}{2}\right)\left(\frac{P_d}{1-\pi_{1,1}}-\frac{P_f}{1-\pi_{1,0}}\right) \\
\Leftrightarrow\ &(N-K^*)\left(\frac{P_d}{1-\pi_{1,1}}-\frac{P_f}{1-\pi_{1,0}}\right) > (K^*-1)\left(\frac{P_d}{\pi_{1,1}}-\frac{P_f}{\pi_{1,0}}\right).
\end{aligned}$$

This completes our proof. □

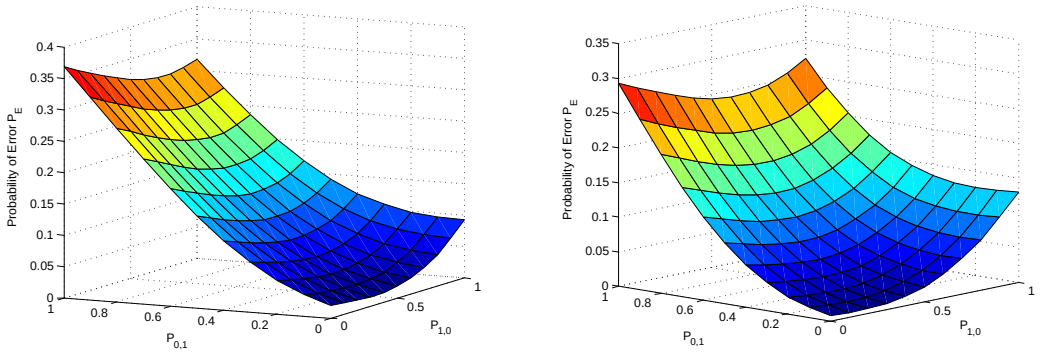
Theorem 3.6.4. $(1,0)$, $(0,1)$, *or* $(1,1)$ *are the optimal attacking strategies* $(P_{1,0}, P_{0,1})$ *that maximize the probability of error* P_E*, when the majority fusion rule is employed at the FC and* $\alpha < \min\{(0.5-P_f), (1-(m/P_d))\}$*, where* $m = \frac{N}{2N-2}$.

Proof. Lemma 3.6.2 and Lemma 3.6.3 suggest that one of the corners is the maximum of P_E because of quasi-convexity. Note that $(0,0)$ cannot be the solution of the maximization problem since the attacker does not flip any results. Hence, we end up with three possibilities: $(1,0)$, $(0,1)$, or $(1,1)$. □

Next, to gain insights into Theorem 3.6.4, we present illustrative examples that corroborate our results.

3.6.1 Illustrative Examples

In Figure 3.3(a), we plot the probability of error P_E as a function of the attacking strategy $(P_{1,0}, P_{0,1})$ for an even number of nodes, $N = 10$, in the network. We assume that the probability of detection is $P_d = 0.8$, the probability of false alarm is $P_f = 0.1$, prior probabilities are $(P_0 = 0.4, P_1 = 0.6)$, and $\alpha = 0.37$. Since $\alpha < \min\{(0.5-P_f), (1-(m/P_d))\}$, where $m = \frac{N}{2N-2}$, quasi-convexity can be observed in Figure 3.3(a). Figure 3.3(b) shows the probability of error P_E as a function of attacking strategy $(P_{1,0}, P_{0,1})$ for odd number of nodes, $N = 11$, in the network. Similarly, quasi-convexity can be observed in Figure 3.3(b).



(a) (b)

Fig. 3.3: (a) P_E as a function of $(P_{1,0}, P_{0,1})$ for $N = 10$. (b) P_E as a function of $(P_{1,0}, P_{0,1})$ for $N = 11$.

It is evident from Figures 3.3(a) and 3.3(b) that the optimal attacking strategy $(P_{1,0}, P_{0,1})$ is either of the following three possibilities: $(1, 0)$, $(0, 1)$, or $(1, 1)$. These results corroborate our theoretical results presented in Theorem 3.6.4.

Observe that the results obtained for this case are not the same as the results obtained for the asymptotic case (Please see Theorem 3.4.2). This is because the asymptotic performance measure (i.e., Chernoff information) is the exponential decay rate of the error probability of the "optimal detector". In other words, while optimizing over Chernoff information, one implicitly assumed that the optimal fusion rule is used at the FC.

Next, we investigate the case where the FC has the knowledge of attacker's strategies and uses the optimal fusion rule K^* to make the global decision. Here, the attacker tries to maximize its worst case probability of error $\min_K P_E$ by choosing $(P_{1,0}, P_{0,1})$ optimally.

3.7 Optimal Byzantine Attacking Strategies with Strategy-aware FC

In this section, we analyze the scenario where the FC has the knowledge of attacker's strategies and uses the optimal fusion rule K^* to make the global decision. The Byzantine attack problem

can be formally stated as follows:

$$\begin{aligned} \underset{P_{j,0},P_{j,1}}{\text{maximize}} \quad & P_E(K^*, \alpha, P_{j,0}, P_{j,1}) \\ \text{subject to} \quad & 0 \leq P_{j,0} \leq 1 \\ & 0 \leq P_{j,1} \leq 1, \end{aligned} \tag{3.40}$$

where K^* is the optimal fusion rule. In other words, K^* is the best response of the FC to the Byzantine attacking strategies. Next, we find the expression for the optimal fusion rule K^* used at the FC.

3.7.1 Optimal Fusion Rule

First, we design the optimal fusion rule assuming that the local sensor threshold λ and the Byzantine attacking strategy $(\alpha, P_{1,0}, P_{0,1})$ are fixed and known to the FC.

Lemma 3.7.1. *For a fixed local sensor threshold λ and $\alpha < \dfrac{1}{P_{0,1} + P_{1,0}}$, the optimal fusion rule is given by*

$$K^* \underset{H_0}{\overset{H_1}{\gtrless}} \frac{\ln\left[(P_0/P_1)\left\{(1-\pi_{1,0})/(1-\pi_{1,1})\right\}^N\right]}{\ln\left[\{\pi_{1,1}(1-\pi_{1,0})\}/\{\pi_{1,0}(1-\pi_{1,1})\}\right]}. \tag{3.41}$$

Proof. Consider the maximum a *posteriori* probability (MAP) rule

$$\frac{P(\mathbf{u}|H_1)}{P(\mathbf{u}|H_0)} \underset{H_0}{\overset{H_1}{\gtrless}} \frac{P_0}{P_1}.$$

Since the u_is are independent of each other, the MAP rule simplifies to

$$\prod_{i=1}^{N} \frac{P(u_i|H_1)}{P(u_i|H_0)} \underset{H_0}{\overset{H_1}{\gtrless}} \frac{P_0}{P_1}.$$

Let us assume that K^* out of N nodes send $u_i = 1$. Now, the above equation can be written as

$$\frac{\pi_{1,1}^{K^*}(1-\pi_{1,1})^{N-K^*}}{\pi_{1,0}^{K^*}(1-\pi_{1,0})^{N-K^*}} \underset{H_0}{\overset{H_1}{\gtrless}} \frac{P_0}{P_1}.$$

Taking logarithms on both sides of the above equation, we have

$$\begin{aligned}
& K^* \ln \pi_{1,1} + (N-K^*)\ln(1-\pi_{1,1}) - K^* \ln \pi_{1,0} - (N-K^*)\ln(1-\pi_{1,0}) \underset{H_0}{\overset{H_1}{\gtrless}} \ln \frac{P_0}{P_1} \\
\Leftrightarrow \quad & K^*[\ln(\pi_{1,1}/\pi_{1,0}) + \ln((1-\pi_{1,0})/(1-\pi_{1,1}))] \underset{H_0}{\overset{H_1}{\gtrless}} \ln \frac{P_0}{P_1} + N\ln((1-\pi_{1,0})/(1-\pi_{1,1})) \\
\Leftrightarrow \quad & K^* \underset{H_0}{\overset{H_1}{\gtrless}} \frac{\ln \frac{P_0}{P_1} + N\ln((1-\pi_{1,0})/(1-\pi_{1,1}))}{[\ln(\pi_{1,1}/\pi_{1,0}) + \ln((1-\pi_{1,0})/(1-\pi_{1,1}))]} \qquad (3.42) \\
\Leftrightarrow \quad & K^* \underset{H_0}{\overset{H_1}{\gtrless}} \frac{\ln\left[(P_0/P_1)\left\{(1-\pi_{1,0})/(1-\pi_{1,1})\right\}^N\right]}{\ln\left[\{\pi_{1,1}(1-\pi_{1,0})\}/\{\pi_{1,0}(1-\pi_{1,1})\}\right]},
\end{aligned}$$

where (3.42) follows from the fact that, for $\pi_{1,1} > \pi_{1,0}$ or equivalently, $\alpha < \dfrac{1}{P_{0,1}+P_{1,0}}$, $[\ln(\pi_{1,1}/\pi_{1,0}) + \ln((1-\pi_{1,0})/(1-\pi_{1,1}))] > 0$. □

The probability of false alarm Q_F and the probability of detection Q_D for this case are as given in (3.1) and (3.2) with $K = \lceil K^* \rceil$. Next, we present our results for the case when the fraction of Byzantines $\alpha > \dfrac{1}{P_{0,1}+P_{1,0}}$.

Lemma 3.7.2. *For a fixed local sensor threshold λ and $\alpha > \dfrac{1}{P_{0,1}+P_{1,0}}$, the optimal fusion rule is given by*

$$K^* \underset{H_1}{\overset{H_0}{\gtrless}} \frac{\ln\left[(P_1/P_0)\left\{(1-\pi_{1,1})/(1-\pi_{1,0})\right\}^N\right]}{[\ln(\pi_{1,0}/\pi_{1,1}) + \ln((1-\pi_{1,1})/(1-\pi_{1,0}))]}. \qquad (3.43)$$

Proof. This can be proved similarly as Lemma 3.7.1 and using the fact that, for $\pi_{1,1} < \pi_{1,0}$ or equivalently, $\alpha > \dfrac{1}{P_{0,1}+P_{1,0}}$, $[\ln(\pi_{1,0}/\pi_{1,1}) + \ln((1-\pi_{1,1})/(1-\pi_{1,0}))] > 0$. □

The probability of false alarm Q_F and the probability of detection Q_D for this case can be calculated to be

$$Q_F = \sum_{i=0}^{\lfloor K^* \rfloor} \binom{N}{i} (\pi_{1,0})^i (1-\pi_{1,0})^{N-i} \qquad (3.44)$$

and

$$Q_D = \sum_{i=0}^{\lfloor K^* \rfloor} \binom{N}{i} (\pi_{1,1})^i (1-\pi_{1,1})^{N-i}. \qquad (3.45)$$

Next, we analyze the property of P_E with respect to Byzantine attacking strategy $(P_{1,0}, P_{0,1})$ that enables us to find the optimal attacking strategies.

Lemma 3.7.3. *For a fixed local sensor threshold λ, assume that the FC employs the optimal fusion rule $\lceil K^* \rceil$, [5] as given in* (3.41)*. Then, for $\alpha \leq 0.5$, the error probability P_E at the FC is a monotonically increasing function of $P_{1,0}$ while $P_{0,1}$ remains fixed. Conversely, the error probability P_E at the FC is a monotonically increasing function of $P_{0,1}$ while $P_{1,0}$ remains fixed.*

Proof. Observe that, for a fixed λ, $P_E(\lceil K^* \rceil)$ is a continuous but not a differentiable function. However, the function is non differentiable only at a finite number (or infinitely countable number) of points because of the nature of $\lceil K^* \rceil$. Now observe that, for a fixed fusion rule K, $P_E(K)$ is differentiable. Utilizing this fact, to show that the lemma is true, we first find the condition that a fusion rule K should satisfy so that P_E is a monotonically increasing function of $P_{1,0}$ while keeping $P_{0,1}$ fixed (and vice versa) and later show that $\lceil K^* \rceil$ satisfies this condition. From (3.25), finding those K that satisfy $\dfrac{dP_E}{dP_{1,0}} > 0$[6] is equivalent to finding those value of K that make

$$\begin{aligned} & r\left(P_{1,0}, K, \alpha\right) > 0 \\ \Leftrightarrow \quad & \ln \frac{P_0}{P_1} \frac{1-P_f}{1-P_d} + (K-1) \ln \frac{\pi_{1,0}}{\pi_{1,1}} + (N-K) \ln \frac{1-\pi_{1,0}}{1-\pi_{1,1}} > 0 \\ \Leftrightarrow \quad & K < \frac{\ln \dfrac{P_0}{P_1} + N \ln \dfrac{(1-\pi_{1,0})}{(1-\pi_{1,1})} + \ln \dfrac{1-P_f}{1-P_d} - \ln \dfrac{\pi_{1,0}}{\pi_{1,1}}}{\ln \left[\{\pi_{1,1}(1-\pi_{1,0})\}/\{\pi_{1,0}(1-\pi_{1,1})\}\right]}. \end{aligned} \tag{3.46}$$

Similarly, we can find the condition that a fusion rule K should satisfy so that P_E is a monotonically increasing function of $P_{0,1}$ while keeping $P_{1,0}$ fixed. From (3.31), finding those K that satisfy

[5] Notice that, K^* might not be an integer.

[6] Observe that, for $\alpha < 0.5$, the function $g\left(P_{1,0}, K^*, \alpha\right) = 0$ (as given in (3.26)) only under extreme conditions (i.e., $P_1 = 0$ or $P_d = 0$ or $P_d = 1$). Ignoring these extreme conditions, we have $g\left(P_{1,0}, K^*, \alpha\right) > 0$.

$\frac{dP_E}{dP_{0,1}} > 0$ is equivalent to finding those K that make

$$\begin{aligned} & r\left(P_{0,1}, K, \alpha\right) > 0 \\ \Leftrightarrow \quad & \ln \frac{P_1}{P_0}\frac{P_d}{P_f} + (K-1)\ln\frac{\pi_{1,1}}{\pi_{1,0}} + (N-K)\ln\frac{1-\pi_{1,1}}{1-\pi_{1,0}} > 0 \\ \Leftrightarrow \quad & K > \frac{\ln\frac{P_0}{P_1} + N\ln\frac{(1-\pi_{1,0})}{(1-\pi_{1,1})} + \ln\frac{P_f}{P_d} - \ln\frac{\pi_{1,0}}{\pi_{1,1}}}{\ln\left[\{\pi_{1,1}(1-\pi_{1,0})\}/\{\pi_{1,0}(1-\pi_{1,1})\}\right]}. \end{aligned} \tag{3.47}$$

From (3.46) and (3.47), we have

$$A = \frac{\ln\frac{P_0}{P_1} + N\ln\frac{(1-\pi_{1,0})}{(1-\pi_{1,1})} + \ln\frac{1-P_f}{1-P_d} - \ln\frac{\pi_{1,0}}{\pi_{1,1}}}{\ln\left[\{\pi_{1,1}(1-\pi_{1,0})\}/\{\pi_{1,0}(1-\pi_{1,1})\}\right]} > K > \frac{\ln\frac{P_0}{P_1} + N\ln\frac{(1-\pi_{1,0})}{(1-\pi_{1,1})} + \ln\frac{P_f}{P_d} - \ln\frac{\pi_{1,0}}{\pi_{1,1}}}{\ln\left[\{\pi_{1,1}(1-\pi_{1,0})\}/\{\pi_{1,0}(1-\pi_{1,1})\}\right]} = B. \tag{3.48}$$

Next, we show that the optimal fusion rule $\lceil K^* \rceil$ given in (3.41) is within the region (A, B). First we prove that $\lceil K^* \rceil > B$ by showing $K^* > B$. Comparing K^* given in (3.41) with B, $K^* > B$ iff

$$0 > \ln\frac{P_f}{P_d} - \ln\frac{\pi_{1,0}}{\pi_{1,1}}. \tag{3.49}$$

Since $P_d > P_f$, to prove (3.49) we start from the inequality

$$\begin{aligned} & \frac{(1-P_d)}{P_d} < \frac{(1-P_f)}{P_f} \\ \Leftrightarrow \quad & \frac{\alpha P_{1,0}(1-P_d) + P_d(1-P_{0,1}\alpha)}{P_d} < \frac{\alpha P_{1,0}(1-P_f) + P_f(1-P_{0,1}\alpha)}{P_f} \\ \Leftrightarrow \quad & \frac{\pi_{1,1}}{P_d} < \frac{\pi_{1,0}}{P_f} \\ \Leftrightarrow \quad & 0 > \ln\frac{P_f}{P_d} - \ln\frac{\pi_{1,0}}{\pi_{1,1}}. \end{aligned}$$

Now, we show that $A > \lceil K^* \rceil$. Observe that,

$$\begin{aligned} & A > \lceil K^* \rceil \\ \Leftrightarrow \quad & \frac{\ln \dfrac{1-P_f}{1-P_d} - \ln \dfrac{\pi_{1,0}}{\pi_{1,1}}}{\ln \left[\{\pi_{1,1}(1-\pi_{1,0})\} / \{\pi_{1,0}(1-\pi_{1,1})\} \right]} > \lceil K^* \rceil - K^*. \end{aligned}$$

Hence, it is sufficient to show that

$$\frac{\ln \dfrac{1-P_f}{1-P_d} - \ln \dfrac{\pi_{1,0}}{\pi_{1,1}}}{\ln \left[\{\pi_{1,1}(1-\pi_{1,0})\} / \{\pi_{1,0}(1-\pi_{1,1})\} \right]} > 1 > \lceil K^* \rceil - K^*.$$

$1 > \lceil K^* \rceil - K^*$ is true from the property of the ceiling function. By (A.27), we have

$$\begin{aligned} & \frac{1-P_f}{1-P_d} > \frac{1-\pi_{1,0}}{1-\pi_{1,1}} \\ \Leftrightarrow \quad & \ln \frac{1-P_f}{1-P_d} > \ln \frac{1-\pi_{1,0}}{1-\pi_{1,1}} \\ \Leftrightarrow \quad & \ln \frac{1-P_f}{1-P_d} - \ln \frac{\pi_{1,0}}{\pi_{1,1}} > \ln \left[\{\pi_{1,1}(1-\pi_{1,0})\} / \{\pi_{1,0}(1-\pi_{1,1})\} \right] \\ \Leftrightarrow \quad & \frac{\ln \dfrac{1-P_f}{1-P_d} - \ln \dfrac{\pi_{1,0}}{\pi_{1,1}}}{\ln \left[\{\pi_{1,1}(1-\pi_{1,0})\} / \{\pi_{1,0}(1-\pi_{1,1})\} \right]} > 1 \end{aligned}$$

which completes the proof. □

Based on Lemma 3.7.3, we present the optimal attacking strategies for the case when the FC has the knowledge regarding the strategies used by the Byzantines.

Theorem 3.7.4. *The optimal attacking strategies, $(P^*_{1,0}, P^*_{0,1})$, which maximize the probability of error, $P_E(\lceil K^* \rceil)$, are given by*

$$(P^*_{1,0}, P^*_{0,1}) \begin{cases} (p_{1,0}, p_{0,1}) & \textit{if } \alpha > 0.5 \\ (1,1) & \textit{if } \alpha \leq 0.5 \end{cases}$$

where $(p_{1,0}, p_{0,1})$ satisfies $\alpha(p_{1,0} + p_{0,1}) = 1$.

Proof. Note that, the maximum probability of error occurs when the posterior probabilities are equal to the prior probabilities of the hypotheses. That is,

$$P(H_i|\mathbf{u}) = P(H_i) \text{ for } i = 0, 1. \tag{3.50}$$

Now using the result from (3.8), the condition can be simplified to

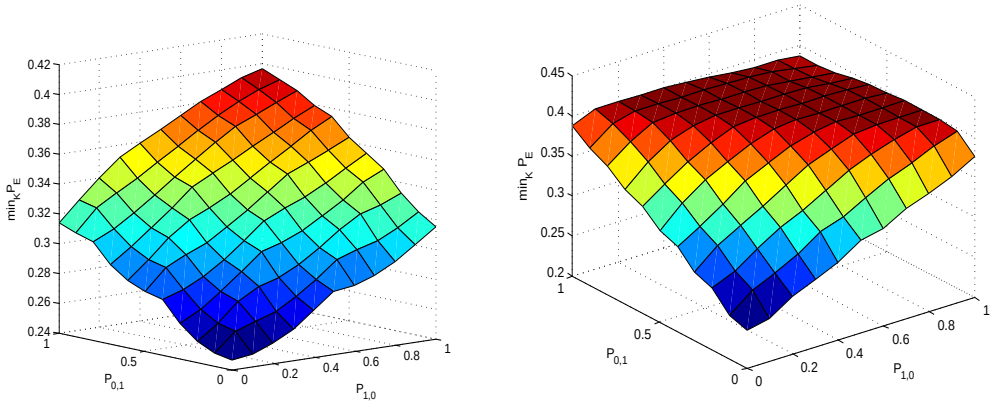
$$\alpha(P_{1,0} + P_{0,1}) = 1. \tag{3.51}$$

Eq. (3.51) suggests that when $\alpha \geq 0.5$, the attacker can find flipping probabilities that make $P_E = \min\{P_0, P_1\}$. When $\alpha = 0.5$, $P_{1,0} = P_{0,1} = 1$ is the optimal attacking strategy and when $\alpha > 0.5$, any pair which satisfies $P_{1,0} + P_{0,1} = \dfrac{1}{\alpha}$ is optimal. However, when $\alpha < 0.5$, (3.51) cannot be satisfied. In this case, by Lemma 3.7.3, for $\alpha < 0.5$, $(1, 1)$ is an optimal attacking strategy, $(P_{1,0}, P_{0,1})$, which maximizes probability of error, $P_E(\lceil K^* \rceil)$. □

Next, to gain insight into Theorem 3.7.4, we present illustrative examples that corroborate our results.

3.7.2 Illustrative Examples

In Figure 3.4, we plot the minimum probability of error as a function of attacker's strategy $(P_{1,0}, P_{0,1})$, where P_E is minimized over all possible fusion rules K. We consider a $N = 11$ node network, with the nodes' detection and false alarm probabilities being 0.6 and 0.4, respectively. Prior probabilities are assumed to be $P_0 = 0.4$ and $P_1 = 0.6$. Observe that, the optimal fusion rule as given in (3.41) changes with attacker's strategy $(P_{1,0}, P_{0,1})$. Thus, the minimum probability of error $\min_K P_E$ is a non-differentiable function. It is evident from Figure 3.4(a) that $(P_{1,0}, P_{0,1}) = (1, 1)$ maximizes the probability of error, $P_E(\lceil K^* \rceil)$. This corroborates our theoretical results presented in Theorem 3.7.4, that for $\alpha < 0.5$, the optimal attacking strategy, $(P_{1,0}, P_{0,1})$, that maximizes the probability of error, $P_E(\lceil K^* \rceil)$, is $(1, 1)$.



(a) (b)

Fig. 3.4: Minimum probability of error ($\min_K P_E$) analysis. (a) $\min_K P_E$ as a function of $(P_{1,0}, P_{0,1})$ for $\alpha = 0.4$. (b) $\min_K P_E$ as a function of $(P_{1,0}, P_{0,1})$ for $\alpha = 0.8$.

In Figure 3.4(b) we consider the scenario where $\alpha = 0.8$ (i.e., $\alpha > 0.5$). It can be seen that the attacking strategy $(P_{1,0}, P_{0,1})$, that maximizes $\min_K P_E$ is not unique in this case. It can be verified that any attacking strategy which satisfies $P_{1,0} + P_{0,1} = \frac{1}{0.8}$ will make $\min_K P_E = \min\{P_0, P_1\} = 0.4$. This corroborates our theoretical results presented in Theorem 3.7.4.

Observe that the results obtained for this case are consistent with the results obtained for the asymptotic case. This is because the optimal fusion rule is used at the FC and the asymptotic performance measure (i.e., Chernoff information) is the exponential decay rate of error probability of the "optimal detector", and thus, implicitly assumes that the optimal fusion rule is used at the FC.

When the attacker does not have the knowledge of the fusion rule K used at the FC, from an attacker's perspective, maximizing its local probability of error P_e is the optimal attacking strategy. The optimal attacking strategy in this case is either of the three possibilities: $(P_{1,0}, P_{0,1}) = (0, 1)$ or $(1, 0)$ or $(1, 1)$ (see Table 3.2). However, the FC has knowledge of the attacking strategy $(\alpha, P_{1,0}, P_{0,1})$ and thus, uses the optimal fusion rule as given in (3.41) and (3.43).

3.8 Joint Optimization of Fusion Rule and Sensor Threshold

In this section, we present a procedure to find the optimal fusion rule and local sensor threshold pair (K^*, λ^*) that minimizes the probability of error P_E given a fixed Byzantine strategy α when $(P_{1,0}, P_{0,1}) = (1, 1)$. This scheme is particularly important in the scenario where Byzantine attackers are performing a man-in-the-middle attack and do not have access to the local sensor threshold. We first show that when using the optimal fusion rule (K^*), P_E is a quasi-convex function of the local sensor threshold (λ) under a certain condition.

Lemma 3.8.1. *For the optimal K and any fixed α ($\alpha < 0.5$), P_E is a quasi-convex function of λ, if $(d/d\lambda)\left(\lambda^{-1}P_d/P_{fa}\right) \leq 0$.*[7]

Proof. A function $f(\lambda)$ is quasi-convex if, for some λ^*, $f(\lambda)$ is non-increasing for $\lambda \leq \lambda^*$ and $f(\lambda)$ is non-decreasing for $\lambda \geq \lambda^*$. In other words, the lemma is proved if $dP_E/d\lambda \leq 0$ (or $dP_E/d\lambda \geq 0$) for all λ, or if for some λ^*, $dP_E/d\lambda \leq 0$ when $\lambda \leq \lambda^*$ and $dP_E/d\lambda \geq 0$ when $\lambda \geq \lambda^*$. Hence, we calculate the partial derivative of P_E with respect to λ. Using the property of ROC's that $dP_d/dP_{fa} = \lambda$, the fact that $d\pi_{11}/d\pi_{10} = \lambda$, we get

$$\begin{aligned}
\frac{dP_E}{d\lambda} &= \pi_0 \frac{dQ_F}{d\lambda} - \pi_1 \frac{dQ_D}{d\lambda} \\
&= -\pi_1 \lambda \left(\pi_{10}\right)' N \begin{pmatrix} N-1 \\ K-1 \end{pmatrix} \left(\pi_{11}\right)^{K-1} \left(1-\pi_{11}\right)^{N-K} \\
&\quad + \pi_0 \left(\pi_{10}\right)' N \begin{pmatrix} N-1 \\ K-1 \end{pmatrix} \left(\pi_{10}\right)^{K-1} \left(1-\pi_{10}\right)^{N-K}
\end{aligned} \tag{3.52}$$

where, $(\pi_{10})' = d\pi_{10}/d\lambda = (1-2\alpha)[dP_{fa}/d\lambda] \leq 0$. The inequality follows from the fact that $x \leq 0.5$ and $dP_{fa}/d\lambda \leq 0$. Following an approach similar to [125], [71], we rewrite the above equation as follows.

$$\frac{dP_E}{d\lambda} = g\left(\lambda, K, \alpha\right)\left(e^{r(\lambda,K,\alpha)} - 1\right) \tag{3.53}$$

[7] Various noise distributions satisfy $(d/d\lambda)\left(\lambda^{-1}P_d/P_{fa}\right) \leq 0$ [71].

where

$$g = N \begin{pmatrix} N-1 \\ K-1 \end{pmatrix} \pi_0(-\pi_{10})'(\pi_{10})^{K-1}(1-\pi_{10})^{N-K} \tag{3.54}$$

and

$$r = \ln\left(\frac{\lambda\pi_1}{\pi_0}\left(\frac{\pi_{11}}{\pi_{10}}\right)^{(K-1)}\left(\frac{1-\pi_{11}}{1-\pi_{10}}\right)^{(N-K)}\right) \tag{3.55}$$

Now it can be seen that $g(\lambda, K, \alpha) \geq 0$. This implies that the sign of $dP_E/d\lambda$ depends on the value of $r(\lambda, K, \alpha)$. The proof is complete if we show that $r(\lambda, K, x)$ is either always positive or negative, or there exists a λ^* such that $r(\lambda, K, \alpha) \leq 0$ for all $\lambda \leq \lambda^*$ and $r(\lambda, K, x) \geq 0$ for all $\lambda \geq \lambda^*$. Substituting $K^* = \frac{\ln\left[(\pi_0/\pi_1)\{(1-\pi_{10})/(1-\pi_{11})\}^N\right]}{\ln[\{\pi_{11}(1-\pi_{10})\}/\{\pi_{10}(1-\pi_{11})\}]}$ in equation (3.55), and dropping K from $r(\lambda, K, x)$ for ease of notation we get $r(\lambda, \alpha) = \ln\lambda - \ln(\pi_{11}/\pi_{10})$. Differentiating $r(\lambda, \alpha)$ with respect to λ, we get

$$\frac{dr(\lambda,\alpha)}{d\lambda} = \frac{1}{\lambda} + \frac{1}{\pi_{11}}\left[\frac{\pi_{11}}{\pi_{10}} - \lambda\right]\frac{d\pi_{10}}{d\lambda} \tag{3.56}$$

In the following, we show that $r(\cdot)$ is non-decreasing. Substituting π_{11}, π_{10}, $d\pi_{10}/d\lambda$ in $dr(\lambda,\alpha)/d\lambda \geq 0$,

$$\frac{\alpha}{1-2\alpha} + P_{fa} + \frac{P_{fa} + \alpha/(1-2\alpha)}{P_d + \alpha/(1-2\alpha)}\left(-\lambda^2\frac{dP_{fa}}{d\lambda}\right) \geq -\lambda\frac{dP_{fa}}{d\lambda},$$

and $P_{fa}/P_d \leq (P_{fa} + \alpha/(1-2\alpha))/(P_d + \alpha/(1-2\alpha))$ since $P_{fa}/P_d \leq 1$. Therefore, it suffices to show that

$$P_{fa} + \lambda\left(-\lambda\frac{dP_{fa}}{d\lambda}\right)\frac{P_{fa}}{P_d} \geq -\lambda\frac{dP_{fa}}{d\lambda}. \tag{3.57}$$

The inequality above is equivalent to the condition in the lemma. □

From (3.53), it can be seen that if $r(K, \lambda^*, \alpha) = 0$ for some λ^* then $(P_E)' = 0$ at λ^* and because P_E is quasi-convex for the optimal fusion rule K^*, it is minimized for $\lambda = \lambda^*$. For the optimal fusion rule K^*, $r(K, \lambda, \alpha) = 0$ has a unique positive root and there exist efficient algorithms, which utilize the quasi-convex nature of the problem, to find an optimum (K^*, λ^*) pair that minimizes P_e [80].

3.9 Discussion

We considered the problem of distributed Bayesian detection with Byzantine data, and characterized the power of attack analytically. For distributed detection for a binary hypothesis testing problem, the expression for the minimum attacking power above which the ability to detect is completely destroyed was obtained. We showed that when there are more than 50% of Byzantines in the network, the data fusion scheme becomes blind and no detector can achieve any performance gain over the one based just on priors. The optimal attacking strategies for Byzantines that degrade the performance at the FC were obtained. It was shown that the results obtained for the non-asymptotic case are consistent with the results obtained for the asymptotic case only when the FC has the knowledge of the attacker's strategies, and thus, uses the optimal fusion rule. However, results obtained for the non-asymptotic case, when the FC does not have knowledge of attacker's strategies, are not the same as the results obtained for the asymptotic case.

CHAPTER 4
DISTRIBUTED DETECTION WITH UNLABELED BYZANTINE DATA: TREE TOPOLOGY

4.1 Introduction

In the previous chapter, the problem of distributed detection in parallel topology was discussed in the presence of Byzantine attacks and the optimal attack strategies were analyzed for the attacker who intends to deteriorate the performance of the detection task at the FC. Even though the parallel topology has received significant attention, there are many practical situations where parallel topology cannot be implemented due to several factors, such as, the FC being outside the communication range of the nodes and limited energy budget of the nodes [60]. In such cases, a multi-hop network is employed, where nodes are organized hierarchically into multiple levels (tree networks). With intelligent use of resources across levels, tree networks have the potential to provide a suitable balance between cost, coverage, functionality, and reliability. Some examples of tree networks include wireless sensor and military communication networks. For instance, the IEEE 802.15.4 (Zigbee) specifications [3] and IEEE 802.22b [40] can support tree-based topolo-

gies.

In this chapter, we address the problem of distributed detection in perfect a-ary tree networks in the presence of Byzantine attacks (data falsification attacks). Well structured (or regular) topologies such as a-ary tree topologies are commonly picked by network designers for their simplicity and, therefore, easier network management. For some practical examples of such networks, one may refer to [61] (and references within). Perfect a-ary tree topologies are widely used in peer to peer systems [53]. Also notice that, designing optimal tree topology for various performance metrics is computationally not feasible. In such scenarios, perfect a-ary topologies provide mathematical tractability and valuable insights into the solution. For previous works on perfect a-ary tree networks, please see [34], [42], [77]. We assume that the cost of attacking nodes at different levels is different and analyze the problem under this assumption. We obtain the expression for minimum attacking power required by the Byzantines to blind the fusion center (FC). More specifically, we show that when more than a certain fraction of individual node decisions are falsified, the decision fusion scheme becomes completely incapable. We also look at the problem from the network designer's (FC) perspective. More specifically, we formulate the robust tree topology design problem as a bi-level program and provide an efficient algorithm to solve it, which is guaranteed to find an optimal solution, if one exists.

The rest of the chapter is organized as follows. Section 4.2 introduces our system model. In Section 4.3, we study the problem from Byzantine's perspective and provide closed form expressions for optimal attacking strategies. In Section 4.4, we formulate the robust topology design problem as a bi-level program and provide an efficient algorithm to solve it in polynomial time. Finally, Section 4.5 concludes the chapter.

4.2 System Model

We consider a distributed detection system with the topology of a perfect a-tree $T(K,\ a)$ rooted at the FC. A perfect a-tree is an a-ary tree in which all the leaf nodes are at the same depth

and all the internal nodes have degree 'a'. $T(K,\ a)$ has a set $\mathcal{N} = \{\mathbb{N}_k\}_{k=1}^{K}$ of transceiver nodes, where $|\mathbb{N}_k| = N_k = a^k$ is the total number of nodes at level (or depth) k. We assume that the depth of the tree is $K > 1$ and the number of children is $a \geq 2$. The total number of nodes in the network is denoted as $\sum_{k=1}^{K} N_k = N$. $\mathcal{B} = \{\mathbb{B}_k\}_{k=1}^{K}$ denotes the set of Byzantine nodes with $|\mathbb{B}_k| = B_k$, where $\mathbb{B}_k$ is the set of Byzantines at level k. The set containing the number of Byzantines residing at levels $1 \leq k \leq K$ is defined as an attack configuration, i.e., $\{B_k\}_{k=1}^{K} = \{|\mathbb{B}_k|\}_{k=1}^{K}$. Notice that, for the attack configuration $\{B_k\}_{k=1}^{K}$, the total number of corrupted paths (or paths containing Byzantine nodes) from Level k to the FC are $\sum_{i=1}^{k} B_i \frac{N_k}{N_i}$, where $B_i \frac{N_k}{N_i}$ gives the total number of covered[1] nodes at level k by B_i Byzantines at level i. If we denote $\alpha_k = \frac{B_k}{N_k}$, then, $\frac{\sum_{i=1}^{k} B_i \frac{N_k}{N_i}}{N_k} = \sum_{i=1}^{k} \alpha_i$ is the fraction of decisions coming from Level k that encounter a Byzantine. In practice, nodes operate with very limited energy and, therefore, it is reasonable to assume that the packet IDs (or source IDs) are not forwarded in the tree to save energy. Moreover, even in cases where the packet IDs (or source IDs) are forwarded, notice that the packet IDs (or source IDs) can be tempered too, thereby preventing the FC to be deterministically aware of the source of a message. Therefore, we consider that the FC looks at messages coming from nodes in a probabilistic manner and considers each received bit to originate from nodes at level k with certain probability $\beta_k \in [0, 1]$. This also implies that, from the FC's perspective, received bits are identically distributed. For a $T(K,\ a)$,

$$\beta_k = \frac{a^k}{N}.$$

4.2.1 Distributed detection in a tree topology

We consider a binary hypothesis testing problem with the two hypotheses H_0 (signal is absent) and H_1 (signal is present). Each node i at level k acts as a source in that it makes a one-bit local decision $v_{k,i} \in \{0, 1\}$ and sends $u_{k,i}$ to its parent node at level $k-1$, where $u_{k,i} = v_{k,i}$ if i is an

[1]Node i at level k' covers all its children at levels $k' + 1 \leq k \leq K$ and the node i itself and, therefore, the total number of covered nodes by $B_{k'}$, Byzantine at level k', is $\frac{B_{k'}}{N_{k'}} . \sum_{i=k'}^{K} N_i$.

uncompromised (honest) node, but for a compromised (Byzantine) node i, $u_{k,i}$ need not be equal to $v_{k,i}$. It also receives the decisions $u_{k',j}$ of all successors j at levels $k' \in [k+1, K]$, which are forwarded to i by its immediate children. It forwards[2] these received decisions along with $u_{k,i}$ to its parent node at level $k-1$. If node i is a Byzantine, then it might alter these received decisions before forwarding. We assume error-free communication channels between children and the parent nodes. We denote the probabilities of detection and false alarm of a honest node i at level k by $P_d^H = P(v_{k,i} = 1|H_1, i \notin \mathbb{B}_k)$ and $P_{fa}^H = P(v_{k,i} = 1|H_0, i \notin \mathbb{B}_k)$, respectively. Similarly, the probabilities of detection and false alarm of a Byzantine node i at level k are denoted by $P_d^B = P(v_{k,i} = 1|H_1, i \in \mathbb{B}_k)$ and $P_{fa}^B = P(v_{k,i} = 1|H_0, i \in \mathbb{B}_k)$, respectively.

We consider the mathematical model presented in 2.2.3 for the Byzantine attack. If a node is honest, then it forwards its own decision and received decisions without altering them. However, a Byzantine node, in order to undermine the network performance, may alter its decision as well as received decisions from its children prior to transmission. We define the following strategies $P_{j,1}^H$, $P_{j,0}^H$ and $P_{j,1}^B$, $P_{j,0}^B$ ($j \in \{0,1\}$) for the honest and Byzantine nodes, respectively, where $P(x = a|y = b)$ is the probability that a node sends a to its parent when it receives b from its child or its actual decision is b. Furthermore, we assume that if a node (at any level) is a Byzantine then none of its ancestors are Byzantines; otherwise, the effect of a Byzantine due to other Byzantines on the same path may be nullified (e.g., Byzantine ancestor re-flipping the already flipped decisions of its successor). This means that any path from a leaf node to the FC will have at most one Byzantine. Thus, we have, $\sum_{k=1}^{K} \alpha_k \leq 1$ since the average number of Byzantines along any path from a leaf to the root cannot be greater than 1.

The Byzantine attacker always wants to degrade the detection performance at the FC as much as possible; in contrast, the FC wants to maximize the detection performance. In this work, we employ the Kullback-Leibler divergence (KLD) [56] $D(\pi_{j,1}||\pi_{j,0})$ to be the network performance metric that characterizes detection performance.

For a K-level network, distributions of received decisions at the FC z_i, $i = 1, .., N$, under

[2]For example, IEEE 802.16j mandates tree forwarding and IEEE 802.11s standardizes a tree-based routing protocol.

$$P(z_i = j|H_0) = \left[\sum_{k=1}^{K} \beta_k \left(\sum_{i=1}^{k} \alpha_i\right)\right] [P_{j,0}^B(1 - P_{fa}^B) + P_{j,1}^B P_{fa}^B] + \left[\sum_{k=1}^{K} \beta_k \left(1 - \sum_{i=1}^{k} \alpha_i\right)\right] [P_{j,0}^H(1 - P_{fa}^H) + P_{j,1}^H P_{fa}^H] \quad (4.1)$$

$$P(z_i = j|H_1) = \left[\sum_{k=1}^{K} \beta_k \left(\sum_{i=1}^{k} \alpha_i\right)\right] [P_{j,0}^B(1 - P_{d}^B) + P_{j,1}^B P_{d}^B] + \left[\sum_{k=1}^{K} \beta_k \left(1 - \sum_{i=1}^{k} \alpha_i\right)\right] [P_{j,0}^H(1 - P_{d}^H) + P_{j,1}^H P_{d}^H] \quad (4.2)$$

H_0 and H_1 are given by (4.1) and (4.2), respectively. In order to make the analysis tractable, we assume that the network designer attempts to maximize the KLD of each node as seen by the FC. On the other hand, the attacker attempts to minimize the KLD of each node as seen by the FC.

Next, we explore the optimal attacking strategies for the Byzantines that most degrade the detection performance by minimizing KLD.

4.3 Optimal Byzantine Attack

As discussed earlier, the Byzantine nodes attempt to make their KL divergence as small as possible. Since the KLD is always non-negative, Byzantines attempt to choose $P(z = j|H_0)$ and $P(z = j|H_1)$ such that KLD is zero. In this case, an adversary can make the data that the FC receives from the nodes such that no information is conveyed. This is possible when

$$P(z = j|H_0) = P(z = j|H_1) \qquad \forall j \in \{0, 1\}. \quad (4.3)$$

Substituting (4.1) and (4.2) in (4.3), the condition to make the $KLD = 0$ for a K-level network can be expressed as

$$P_{j,1}^B - P_{j,0}^B = \frac{\sum_{k=1}^{K}[\beta_k(1 - \sum_{i=1}^{k} \alpha_i)]}{\sum_{k=1}^{K}[\beta_k(\sum_{i=1}^{k} \alpha_i)]]} \frac{P_d^H - P_{fa}^H}{P_d^B - P_{fa}^B}(P_{j,0}^H - P_{j,1}^H). \quad (4.4)$$

We have

$$P^B_{0,1} - P^B_{0,0} = \frac{\sum_{k=1}^{K}[\beta_k(1-\sum_{i=1}^{k}\alpha_i)]}{\sum_{k=1}^{K}[\beta_k(\sum_{i=1}^{k}\alpha_i)]]}\frac{P^H_d - P^H_{fa}}{P^B_d - P^B_{fa}} = -(P^B_{1,1} - P^B_{1,0}). \tag{4.5}$$

Hence, the attacker can degrade detection performance by intelligently choosing $(P^B_{0,1}, P^B_{1,0})$, which are dependent on α_k, for $k = 1, \cdots, K$. Observe that,

$$0 \leq P^B_{0,1} - P^B_{0,0}$$

since $\sum_{i=1}^{k}\alpha_i \leq 1$ for $k \leq K$. To make $KLD = 0$, we must have

$$P^B_{0,1} - P^B_{0,0} \leq 1$$

such that $(P^B_{j,1}, P^B_{j,0})$ becomes a valid probability mass function. Notice that, when $P^B_{0,1} - P^B_{0,0} > 1$ there does not exist any attacking probability distribution $(P^B_{j,1}, P^B_{j,0})$ that can make $KLD = 0$. In the case of $P^B_{0,1} - P^B_{0,0} = 1$, there exists a unique solution $(P^B_{1,1}, P^B_{1,0}) = (0, 1)$ that can make $KLD = 0$. For the $P^B_{0,1} - P^B_{0,0} < 1$ case, there exist an infinite number of attacking probability distributions $(P^B_{j,1}, P^B_{j,0})$ which can make $KLD = 0$.

By further assuming that the honest and Byzantine nodes are identical in terms of their detection performance, i.e., $P^H_d = P^B_d$ and $P^H_{fa} = P^B_{fa}$, the above condition to blind the FC reduces to

$$\frac{\sum_{k=1}^{K}[\beta_k(1-\sum_{i=1}^{k}\alpha_i)]}{\sum_{k=1}^{K}[\beta_k(\sum_{i=1}^{k}\alpha_i)]]} \leq 1$$

which is equivalent to

$$\sum_{k=1}^{K}[\beta_k(1 - 2(\sum_{i=1}^{k}\alpha_i))] \leq 0. \tag{4.6}$$

Recall that $\alpha_k = \frac{B_k}{N_k}$ and $\beta_k = \frac{N_k}{\sum_{i=1}^{K} N_i}$. Substituting α_k and β_k into (4.6) and simplifying the result, we have the following proposition.

Proposition 4.3.1. *In a tree network with K levels, there exists an attacking probability distribution $(P^B_{0,1}, P^B_{1,0})$ that can make $KLD = 0$, and thereby blind the FC, if and only if $\{B_k\}_{k=1}^{K}$*

satisfy

$$\sum_{k=1}^{K}\left(\frac{B_k}{N_k}\sum_{i=k}^{K}N_i\right)\geq\frac{N}{2}. \tag{4.7}$$

Dividing both sides of (4.7) by N, the above condition can be written as $\sum_{k=1}^{K}\beta_k\sum_{i=1}^{k}\alpha_i\geq 0.5$. This implies that to make the FC blind, 50% or more nodes in the network need to be covered by the Byzantines. Observe that, Proposition 4.3.1 suggests that there exist multiple attack configurations $\{B_k\}_{k=1}^{K}$ that can blind the FC. Also notice that, some of these attacking sets require Byzantines to compromise less than 50% of the nodes in the network. For example, attacking half of the nodes at Level 1 (i.e., $B_1=\frac{N_1}{2}<<\frac{N}{2}$) cover 50% of the nodes in the network and, therefore, the FC becomes blind. This implies that in the tree topology Byzantines have more degrees of freedom to blind the FC as compared to the parallel topology.

Next, to explore the optimal attacking probability distribution $(P_{0,1}^{B},P_{1,0}^{B})$ that minimizes KLD when (4.6) does not hold, we explore the properties of KLD.

First, we show that attacking with symmetric flipping probabilities is the optimal strategy in the region where the attacker cannot blind the FC. In other words, attacking with $P_{1,0}=P_{0,1}$ is the optimal strategy for the Byzantines. For analytical tractability, we assume $P_d^H=P_d^B=P_d$ and $P_{fa}^H=P_{fa}^B=P_{fa}$ in further analysis.

Lemma 4.3.2. *In the region where the attacker cannot blind the FC, the optimal attacking strategy comprises of symmetric flipping probabilities. More specifically, any non zero deviation $\epsilon_i\in(0,p]$ in flipping probabilities $(P_{0,1}^{B},P_{1,0}^{B})=(p-\epsilon_1,p-\epsilon_2)$, where $\epsilon_1\neq\epsilon_2$, will result in increase in the KLD.*

Proof. Let us denote, $P(z=1|H_1)=\pi_{1,1}$, $P(z=1|H_0)=\pi_{1,0}$ and $t=\sum_{k=1}^{K}\beta_k\sum_{i=1}^{k}\alpha_i$. Notice that, in the region where the attacker cannot blind the FC, the parameter $t<0.5$. To prove the lemma, we first show that any positive deviation $\epsilon\in(0,p]$ in flipping probabilities $(P_{1,0}^{B},P_{0,1}^{B})=(p,p-\epsilon)$ will result in an increase in the KLD. After plugging in $(P_{1,0}^{B},P_{0,1}^{B})=$

$(p, p-\epsilon)$ in (4.1) and (4.2), we get

$$\pi_{1,1} = t(p - P_d(2p-\epsilon)) + P_d \tag{4.8}$$

$$\pi_{1,0} = t(p - P_{fa}(2p-\epsilon)) + P_{fa}. \tag{4.9}$$

Now we show that the KLD, D, is a monotonically increasing function of the parameter ϵ or in other words, $\frac{dD}{d\epsilon} > 0$.

$$\begin{aligned}\frac{dD}{d\epsilon} &= \pi_{1,1}\left(\frac{\pi'_{1,1}}{\pi_{1,1}} - \frac{\pi'_{1,0}}{\pi_{1,0}}\right) + \pi'_{1,1}\log\frac{\pi_{1,1}}{\pi_{1,0}} \\ &+ (1-\pi_{1,1})\left(\frac{\pi'_{1,0}}{1-\pi_{1,0}} - \frac{\pi'_{1,1}}{1-\pi_{1,1}}\right) - \pi'_{1,1}\log\frac{1-\pi_{1,1}}{1-\pi_{1,0}}\end{aligned} \tag{4.10}$$

where $\frac{d\pi_{1,1}}{d\epsilon} = \pi'_{1,1} = tP_d$ and $\frac{d\pi_{1,0}}{d\epsilon} = \pi'_{1,0} = tP_{fa}$ and t is the fraction of covered nodes by the Byzantines. After rearranging the terms in the above equation, the condition $\frac{dD}{d\epsilon} > 0$ becomes

$$\frac{1-\pi_{1,1}}{1-\pi_{1,0}} + \frac{P_d}{P_{fa}}\log\frac{\pi_{1,1}}{\pi_{1,0}} > \frac{\pi_{1,1}}{\pi_{1,0}} + \frac{P_d}{P_{fa}}\log\frac{1-\pi_{1,1}}{1-\pi_{1,0}}. \tag{4.11}$$

Since $P_d > P_{fa}$ and $t < 0.5$, $\pi_{1,1} > \pi_{1,0}$. It can also be proved that $\frac{P_{fa}}{P_d}\frac{\pi_{1,1}}{\pi_{1,0}} < 1$. Hence, we have

$$\begin{aligned}& 1 + (\pi_{1,1} - \pi_{1,0}) > \frac{P_{fa}}{P_d}\frac{\pi_{1,1}}{\pi_{1,0}} \\ \Leftrightarrow\ & (\pi_{1,1} - \pi_{1,0})\left[\frac{1 + (\pi_{1,1} - \pi_{1,0})}{\pi_{1,1}(1-\pi_{1,0})}\right] > \frac{P_{fa}}{P_d}\frac{\pi_{1,1}}{\pi_{1,0}}\left[\frac{\pi_{1,1} - \pi_{1,0}}{\pi_{1,1}(1-\pi_{1,0})}\right] \\ \Leftrightarrow\ & \left[\frac{1-\pi_{1,0} - (1-\pi_{1,1})}{1-\pi_{1,0}} + \frac{(\pi_{1,1} - \pi_{1,0})}{\pi_{1,1}}\right] > \frac{P_{fa}}{P_d}\left[\frac{\pi_{1,1}}{\pi_{1,0}} - \frac{1-\pi_{1,1}}{1-\pi_{1,0}}\right] \\ \Leftrightarrow\ & \frac{1-\pi_{1,1}}{1-\pi_{1,0}} + \frac{P_d}{P_{fa}}\left(1 - \frac{\pi_{1,0}}{\pi_{1,1}}\right) > \frac{\pi_{1,1}}{\pi_{1,0}} + \frac{P_d}{P_{fa}}\left(\frac{1-\pi_{1,1}}{1-\pi_{1,0}} - 1\right).\end{aligned} \tag{4.12}$$

To prove that (4.11) is true, we apply the logarithm inequality $(x-1) \geq \log x \geq \frac{x-1}{x}$, for $x > 0$ to (4.12). First, let us assume that $x = \frac{\pi_{1,1}}{\pi_{1,0}}$. Now, using the logarithm inequality we can show that $\log\frac{\pi_{1,1}}{\pi_{1,0}} \geq 1 - \frac{\pi_{1,0}}{\pi_{1,1}}$. Next, let us assume that $x = \frac{1-\pi_{1,1}}{1-\pi_{1,0}}$. Again, using the logarithm inequality

it can be shown that $\left[\frac{1-\pi_{1,1}}{1-\pi_{1,0}} - 1\right] \geq \log \frac{1-\pi_{1,1}}{1-\pi_{1,0}}$. Using these results and (4.12), one can prove that condition (4.11) is true.

Similarly, we can show that any non zero deviation $\epsilon \in (0, p]$ in flipping probabilities $(P^B_{1,0}, P^B_{0,1}) = (p-\epsilon, p)$ will result in an increase in the KLD, i.e., $\frac{dD}{d\epsilon} > 0$, or

$$\frac{\pi_{1,1}}{\pi_{1,0}} + \frac{1-P_d}{1-P_{fa}} \log \frac{1-\pi_{1,1}}{1-\pi_{1,0}} > \frac{1-\pi_{1,1}}{1-\pi_{1,0}} + \frac{1-P_d}{1-P_{fa}} \log \frac{\pi_{1,1}}{\pi_{1,0}}. \tag{4.13}$$

Since $P_d > P_{fa}$ and $t < 0.5$, $\pi_{1,1} > \pi_{1,0}$. It can be proved that $\frac{1-\pi_{1,1}}{1-\pi_{1,0}} > \frac{1-P_d}{1-P_{fa}}$. Hence, we have

$$\frac{1-\pi_{1,1}}{1-\pi_{1,0}} > \frac{1-P_d}{1-P_{fa}} \left[1 - (\pi_{1,1} - \pi_{1,0})\right] \tag{4.14}$$

$$\Leftrightarrow \quad \frac{1-\pi_{1,1}}{\pi_{1,0}(1-\pi_{1,0})} > \frac{1-P_d}{1-P_{fa}} \left[\frac{1-(\pi_{1,1}-\pi_{1,0})}{\pi_{1,0}}\right]$$

$$\Leftrightarrow \quad \frac{1}{\pi_{1,0}(1-\pi_{1,0})} > \frac{1-P_d}{1-P_{fa}} \left[\frac{1-(\pi_{1,1}-\pi_{1,0})}{\pi_{1,0}(1-\pi_{1,1})}\right]$$

$$\Leftrightarrow \quad \frac{1}{\pi_{1,1}-\pi_{1,0}} \left[\frac{\pi_{1,1}}{\pi_{1,0}} - \frac{1-\pi_{1,1}}{1-\pi_{1,0}}\right] > \frac{1-P_d}{1-P_{fa}} \left[\frac{1}{\pi_{1,0}} + \frac{1}{1-\pi_{1,1}}\right] \tag{4.15}$$

$$\Leftrightarrow \quad \frac{\pi_{1,1}}{\pi_{1,0}} - \frac{1-\pi_{1,1}}{1-\pi_{1,0}} > \frac{1-P_d}{1-P_{fa}} \left[\frac{\pi_{1,1}-\pi_{1,0}}{\pi_{1,0}} + \frac{\pi_{1,1}-\pi_{1,0}}{1-\pi_{1,1}}\right] \tag{4.16}$$

$$\Leftrightarrow \quad \frac{\pi_{1,1}}{\pi_{1,0}} + \frac{1-P_d}{1-P_{fa}} \left[1 - \frac{1-\pi_{1,0}}{1-\pi_{1,1}}\right] > \frac{1-\pi_{1,1}}{1-\pi_{1,0}} + \frac{1-P_d}{1-P_{fa}} \left[\frac{\pi_{1,1}}{\pi_{1,0}} - 1\right]. \tag{4.17}$$

To prove that (4.13) is true, we apply the logarithm inequality $(x-1) \geq \log x \geq \frac{x-1}{x}$, for $x > 0$ to (4.17). First, let us assume that $x = \frac{1-\pi_{1,1}}{1-\pi_{1,0}}$. Now, using the logarithm inequality we can show that $\log \frac{1-\pi_{1,1}}{1-\pi_{1,0}} \geq 1 - \frac{1-\pi_{1,0}}{1-\pi_{1,1}}$. Next, let us assume that $x = \frac{\pi_{1,1}}{\pi_{1,0}}$. Again, using the logarithm inequality it can be shown that $\left[\frac{\pi_{1,1}}{\pi_{1,0}} - 1\right] \geq \log \frac{\pi_{1,1}}{\pi_{1,0}}$. Using these results and (4.17), one can prove that condition (4.13) is true. Condition (4.11) and (4.13) imply that any non zero deviation $\epsilon_i \in (0, p]$ in flipping probabilities $(P^B_{0,1}, P^B_{1,0}) = (p-\epsilon_1, p-\epsilon_2)$ will result in an increase in the KLD. □

In the next theorem, we present a closed form expression for the optimal attacking probability distribution $(P^B_{j,1}, P^B_{j,0})$ that minimizes KLD in the region where the attacker cannot blind the FC.

Theorem 4.3.3. *In the region where the attacker cannot blind the FC, the optimal attacking strategy is given by* $(P^B_{0,1}, P^B_{1,0}) = (1, 1)$.

Proof. Observe that, in the region where the attacker cannot blind the FC, the optimal strategy comprises of symmetric flipping probabilities $(P^B_{0,1} = P^B_{1,0} = p)$. The proof is complete if we show that KLD, D, is a monotonically decreasing function of the flipping probability p.

Let us denote, $P(z = 1|H_1) = \pi_{1,1}$ and $P(z = 1|H_0) = \pi_{1,0}$. After plugging in $(P^B_{0,1}, P^B_{1,0}) = (p, p)$ in (4.1) and (4.2), we get

$$\pi_{1,1} = t(p - P_d(2p)) + P_d \tag{4.18}$$

$$\pi_{1,0} = t(p - P_{fa}(2p)) + P_{fa}. \tag{4.19}$$

Now we show that the KLD, D, is a monotonically decreasing function of the parameter p or in other words, $\frac{dD}{dp} < 0$. After plugging in $\pi'_{1,1} = t(1-2P_d)$ and $\pi'_{1,0} = t(1-2P_{fa})$ in the expression of $\frac{dD}{dp}$ and rearranging the terms, the condition $\frac{dD}{dp} < 0$ becomes

$$(1 - 2P_{fa}) \left(\frac{1 - \pi_{1,1}}{1 - \pi_{1,0}} - \frac{\pi_{1,1}}{\pi_{1,0}} \right) + (1 - 2P_d) \log \left(\frac{1 - \pi_{1,0}}{1 - \pi_{1,1}} \frac{\pi_{1,1}}{\pi_{1,0}} \right) < 0 \tag{4.20}$$

Since $P_d > P_{fa}$ and $t < 0.5$, we have $\pi_{1,1} > \pi_{1,0}$. Now, using the fact that $\frac{1 - P_d}{1 - P_{fa}} > \frac{1 - 2P_d}{1 - 2P_{fa}}$ and (4.15), we have

$$\begin{aligned} & \frac{1}{\pi_{1,1} - \pi_{1,0}} \left[\frac{\pi_{1,1}}{\pi_{1,0}} - \frac{1 - \pi_{1,1}}{1 - \pi_{1,0}} \right] > \frac{1 - 2P_d}{1 - 2P_{fa}} \left[\frac{1}{\pi_{1,0}} + \frac{1}{1 - \pi_{1,1}} \right] \\ \Leftrightarrow \quad & \frac{\pi_{1,1}}{\pi_{1,0}} + \frac{1 - 2P_d}{1 - 2P_{fa}} \left[1 - \frac{1 - \pi_{1,0}}{1 - \pi_{1,1}} \right] > \frac{1 - \pi_{1,1}}{1 - \pi_{1,0}} + \frac{1 - 2P_d}{1 - 2P_{fa}} \left[\frac{\pi_{1,1}}{\pi_{1,0}} - 1 \right]. \end{aligned} \tag{4.21}$$

Applying the logarithm inequality $(x - 1) \geq \log x \geq \frac{x - 1}{x}$, for $x > 0$ to (4.21), one can prove that (4.20) is true. □

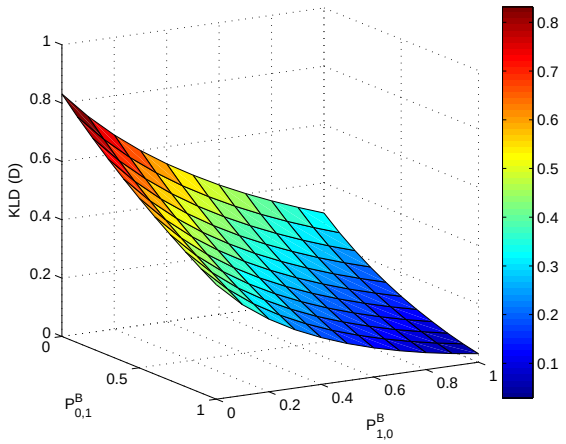


Fig. 4.1: KL distance vs Flipping Probabilities when $P_d = 0.8$, $P_{fa} = 0.2$, and the fraction of covered nodes by the Byzantines is $t = 0.4$

Next, to gain insights into the solution, we present some numerical results in Figure 4.1 that corroborate our theoretical results. We plot KLD as a function of the flipping probabilities $(P^B_{1,0},\ P^B_{0,1})$. We assume that the probability of detection is $P_d = 0.8$, the probability of false alarm is $P_{fa} = 0.2$ and the fraction of covered nodes by the Byzantines is $t = 0.4$. It can be seen that the optimal attacking strategy comprises of symmetric flipping probabilities and is given by $(P^B_{0,1}, P^B_{1,0}) = (1, 1)$, which corroborate our theoretical result presented in Lemma 4.3.2 and Theorem 4.3.3.

Next, we explore some properties of the KLD with respect to the fraction of covered nodes t in the region where the attacker cannot blind the FC, i.e., $t < 0.5$.

Lemma 4.3.4. *$D^* = \min\limits_{(P^B_{j,1},P^B_{j,0})} D(\pi_{j,1}||\pi_{j,0})$ is a continuous, decreasing and convex function of fraction of covered nodes by the Byzantines $t = \sum_{k=1}^{K}[\beta_k(\sum_{i=1}^{k}\alpha_i)]$ in the region where the attacker cannot blind the FC ($t < 0.5$).*

Proof. The continuity of $D(\pi_{j,1}||\pi_{j,0})$ with respect to the involved distributions implies the continuity of D^*. To show that D^* is a decreasing function of t, we use the fact that $\operatorname{argmin}\limits_{(P^B_{0,1},P^B_{1,0})} D(\pi_{j,1}||\pi_{j,0})$ is equal to $(1, 1)$ for $t < 0.5$ (as shown in Theorem 4.3.3). After plugging $(P^B_{0,1}, P^B_{1,0}) = (1, 1)$ in the KLD expression, it can be shown that the expression for the derivative of D with respect to t, $\frac{dD}{dt}$, is the same as (4.20). Using the results of Theorem 4.3.3, it follows that $\frac{dD}{dt} < 0$ and,

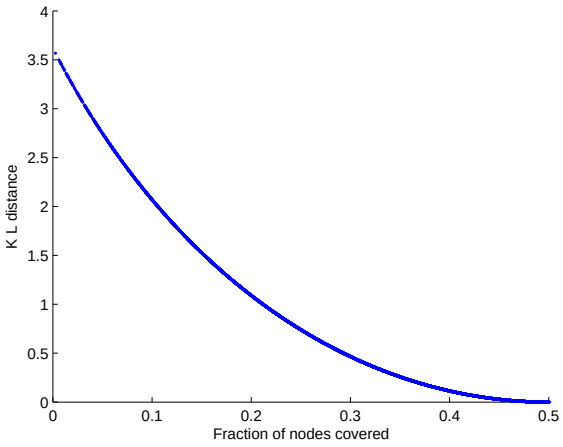


Fig. 4.2: $\min_{(P^B_{j,1},P^B_{j,0})}$ KL distance vs Fraction of nodes covered when $P_d = 0.8$ and $P_{fa} = 0.2$

therefore, D^* is a monotonically decreasing function of t in the region where $t < 0.5$. The convexity of D^* follows from the fact that $D^*(\pi_{j,1}||\pi_{j,0})$ is convex in $\pi_{j,1}$ and $\pi_{j,0}$, which are affine transformations of t (Note that, convexity holds under affine transformation). □

It is worth noting that Lemma 4.3.4 suggests that by minimizing/maximizing the fraction of covered nodes t, the FC can maximize/minimize the KLD. Using this fact, from now onwards we will consider fraction of covered nodes t in lieu of the KLD in further analysis in the chapter.

Next, to gain insights into the solution, we present some numerical results in Figure 4.2 that corroborate our theoretical results. We plot $\min_{(P^B_{j,1},P^B_{j,0})}$ KLD as a function of the fraction of covered nodes. We assume that the probabilities of detection and false alarm are $P_d = 0.8$ and $P_{fa} = 0.2$, respectively. Notice that, when 50% of the nodes in the network are covered, KLD between the two probability distributions becomes zero and FC becomes blind. It can be seen that D^* is a continuous, decreasing and convex function of the fraction of covered nodes t in the region $t < 0.5$, which corroborate our theoretical result presented in Lemma 4.3.4.

Until now, we have explored the problem from the attacker's perspective. In the rest of the chapter, we look into the problem from a network designer's perspective and propose a technique to mitigate the effect of the Byzantines. More specifically, we explore the problem of designing a robust tree topology considering the Byzantine to incur a cost for attacking the network and the FC

to incur a cost for deploying (including the cost of protection, etc.) the network. The FC (network designer) tries to design a perfect a-ary tree topology under its cost budget constraint such that the system performance metric, i.e., KLD is maximized. Byzantines, on the other hand, are interested in attacking or capturing nodes to cause maximal possible degradation in system performance, with the cost of attacking or capturing nodes not to exceed the attacker's budget. This problem can be formulated as a bi-level programming problem where the upper and the lower level problems with conflicting objectives belong to the leader (FC) and the follower (Byzantines), respectively.

4.4 Robust Topology Design

In this problem setting, it is assumed that there is a cost associated with attacking each node in the tree (which may represent resources required for capturing a node or cloning a node in some cases). We also assume that the costs for attacking nodes at different levels are different. Specifically, let c_k be the cost of attacking any one node at level k. Also, we assume $c_k > c_{k+1}$ for $k = 1, \cdots, K-1$, i.e., it is more costly to attack nodes that are closer to the FC. Observe that, a node i at level k covers (in other words, can alter the decisions of) all its successors and node i itself. It is assumed that the network designer or the FC has a cost budget $C_{budget}^{network}$ and the attacker has a cost budget $C_{budget}^{attacker}$. Let P_k denote the number of nodes covered by a node at level k. We refer to P_k as the "profit" of a node at level k. Notice that, $P_k = \frac{\sum_{i=k+1}^{K} N_i}{N_k} + 1$.

Notice that, in a tree topology, P_k can be written as

$$P_k = a_k \times P_{k+1} + 1 \qquad for\ k = 1, ..., K-1, \tag{4.22}$$

where P_k is the profit of attacking a node at level k, P_{k+1} is the profit of attacking a node at level $k+1$ and a_k is the number of immediate children of a node at level k. For a perfect a-ary tree $a_k = a,\ \forall k$ and $P_k = \frac{a^{K-k+1}-1}{a-1}$. The FC designs the network, such that, given the attacker's budget, the fraction of covered nodes is minimized, and consequently a more robust perfect a-ary tree in terms of KLD (See Lemma 4.3.4) is generated.

4.4.1 Robust Perfect a-ary Tree Topology Design

Since the attacker aims to maximize the fraction of covered nodes by attacking/capturing $\{B_k\}_{k=1}^{K}$ nodes within the cost budget $C_{budget}^{attacker}$, the FC's objective is to minimize the fraction of covered nodes by choosing the parameters $(K,\ a)$ optimally in a perfect a-ary tree topology $T(K,\ a)$ under its cost budget $C_{budget}^{network}$. This situation can be interpreted as a Bi-level optimization problem, where the first decision maker (the so-called leader) has the first choice, and the second one (the so-called follower) reacts optimally to the leader's selection. It is the leader's aim to find such a decision which, together with the optimal response of the follower, optimizes the objective function of the leader. For our problem, the upper level problem (ULP) corresponds to the FC who is the leader of the game, while the lower level problem (LLP) belongs to the attacker who is the follower. We assume that the FC has complete information about the attacker's problem, i.e., the objective function and the constraints of the LLP. Similarly, the attacker is assumed to be aware about the FC's resources, i.e., cost of deploying the nodes $\{c_k\}_{k=1}^{K}$. Next, we formalize our robust perfect a-ary tree topology problem as follows:

$$
\begin{aligned}
\underset{(K,\,a)\in\mathbb{Z}^+}{\text{minimize}} \quad & \frac{\sum_{k=1}^{K}(a^{K-k+1}-1)B_k}{a(a^K-1)} \\
\text{subject to} \quad & a_{min} \leq a \leq a_{max} \\
& K \geq K_{min} \\
& \sum_{k=1}^{K} a^k \geq N_{min} \\
& \sum_{k=1}^{K} c_k a^k \leq C_{budget}^{network} \\
& \underset{B_k\in\mathbb{Z}^+}{\text{maximize}} \quad \frac{\sum_{k=1}^{K}(a^{K-k+1}-1)B_k}{a(a^K-1)} \\
& \text{subject to} \quad \sum_{k=1}^{K} c_k B_k \leq C_{budget}^{attacker} \\
& \qquad\qquad B_k \leq a^k, \forall\, k = 1, 2, \ldots, K
\end{aligned}
\tag{4.23}
$$

where $\mathbb{Z}^+$ is the set of non-negative integers, $a_{min} \geq 2$ and $K_{min} \geq 2$. The objective function in ULP is the fraction of covered nodes by the Byzantines $\frac{\sum_{k=1}^{K} P_k B_k}{\sum_{k=1}^{K} N_k}$, where $P_k = \frac{a^{K-k+1}-1}{a-1}$ and $\sum_{k=1}^{K} N_k = \frac{a(a^K-1)}{a-1}$. In the constraint $a_{min} \leq a \leq a_{max}$, a_{max} represents the hardware constraint imposed by the Medium Access Control (MAC) scheme used and a_{min} represents the design constraint enforced by the FC. The constraint on the number of nodes in the network $\sum_{k=1}^{K} a^k \geq N_{min}$ ensures that the network satisfies pre-specified detection performance guarantees. In other words, N_{min} is the minimum number of nodes needed to guarantee a certain detection performance. The constraint on the cost expenditure $\sum_{k=1}^{K} c_k a^k \leq C_{budget}^{network}$ ensures that the total expenditure of the network designer does not exceed the available budget.

In the LLP, the objective function is the same as that of the FC, but the sense of optimization is opposite, i.e., maximization of the fraction of covered nodes. The constraint $\sum_{k=1}^{K} c_k B_k \leq C_{budget}^{attacker}$ ensures that the total expenditure of the attacker does not exceed the available budget. The constraints $B_k \leq a^k, \ \forall k$ are logical conditions, which prevent the attacker from attacking non-existing resources.

Notice that, the bi-level optimization problem, in general, is an NP-hard problem [8]. In fact, the optimization problem corresponding to LLP is the packing formulation of the Bounded Knapsack Problem (BKP) [27], which itself, in general, is NP-hard. Next, we discuss some properties of our objective function that enable our robust topology design problem to have a polynomial time solution.

Lemma 4.4.1. *In a perfect a-ary tree topology, the fraction of covered nodes $\frac{\sum_{k=1}^{K} P_k B_k}{\sum_{k=1}^{K} N_k}$ by the attacker with the cost budget $C_{budget}^{attacker}$ for an optimal attack is a non-decreasing function of the number of levels K in the tree.*

Proof. Let us denote the optimal attack configuration for a K level *perfect a-ary* tree topology $T(K, \ a)$ by $\{B_k^1\}_{k=1}^{K}$ and the optimal attack configuration for a *perfect a-ary* tree topology with $K+1$ levels by $\{B_k^2\}_{k=1}^{K+1}$ given the cost budget $C_{budget}^{attacker}$. To prove the lemma, it is sufficient to show that

$$\frac{\sum_{k=1}^{K+1} P_k^2 B_k^2}{\sum_{k=1}^{K+1} N_k} \geq \frac{\sum_{k=1}^{K} P_k^2 B_k^1}{\sum_{k=1}^{K+1} N_k} \geq \frac{\sum_{k=1}^{K} P_k^1 B_k^1}{\sum_{k=1}^{K} N_k}, \tag{4.24}$$

where P_k^1 is the profit of attacking a node at level k in a K level *perfect* a*-ary* tree topology and P_k^2 is the profit of attacking a node at level k in a $K+1$ level *perfect* a*-ary* tree topology.

First inequality in (4.24) follows due to the fact that $\{B_k^1\}_{k=1}^K$ may not be the optimal attack configuration for topology $T(K+1, a)$. To prove the second inequality observe that, an increase in the value of parameter K results in an increase in both the denominator (number of nodes in the network) and the numerator (fraction of covered nodes). Using this fact, let us denote

$$\frac{\sum_{k=1}^{K} P_k^2 B_k^1}{\sum_{k=1}^{K+1} N_k} = \frac{x + x_1}{y + y_1} \tag{4.25}$$

with $x = \sum_{k=1}^{K} P_k^1 B_k^1$ with $P_k^1 = \frac{a^{K-k+1} - 1}{a-1}$, $y = \sum_{k=1}^{K} N_k = \frac{a(a^K - 1)}{a-1}$, $x_1 = \sum_{k=1}^{K} (B_k^1 a^{K-k+1})$ is the increase in the profit by adding one more level to the topology and $y_1 = a^{K+1}$ is the increase in the number of nodes in the network by adding one more level to the topology .

Note that $\frac{x + x_1}{y + y_1} > \frac{x}{y}$ if and only if

$$\frac{x}{y} < \frac{x_1}{y_1} \tag{4.26}$$

where x, y, x_1, and y_1 are positive values. Hence, it is sufficient to prove that

$$\frac{a^{K+1} \sum_{k=1}^{K} \left(\frac{B_k^1}{a^k}\right) - \sum_{k=1}^{K} B_k^1}{a(a^K - 1)} \leq \frac{\sum_{k=1}^{K} (B_k^1 a^{K-k+1})}{a^{K+1}}.$$

The above equation can be further simplified to

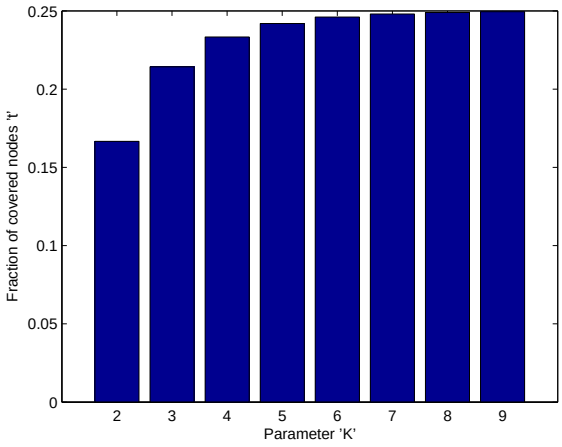


Fig. 4.3: Fraction of nodes covered vs Parameter K when $a = 2$, K is varied from 2 to 9, $[c_1, \cdots, c_9] = [52,\ 48,\ 24,\ 16,\ 12,\ 8,\ 10,\ 6,\ 4]$, and $C_{budget}^{attacker} = 50$

$$\sum_{k=1}^{K} \left(\frac{B_k^1}{a^k} \right) \leq \sum_{k=1}^{K} \left(\frac{B_k^1}{a} \right)$$

which is true for all $K \geq 1$. □

Next, to gain insights into the solution, we present some numerical results in Figure 4.3 that corroborate our theoretical results. We plot the fraction of covered nodes by the Byzantines as a function of the total number of levels in the tree. We assume that $a = 2$ and vary K from 2 to 9. We also assume that the cost to attack nodes at different levels are given by $[c_1, \cdots, c_9] = [52,\ 48,\ 24,\ 16,\ 12,\ 8,\ 10,\ 6,\ 4]$ and the cost budget of the attacker is $C_{budget}^{attacker} = 50$. For each $T(K,\ 2)$, we find the optimal attack configuration $\{B_k\}_{k=1}^{K}$ by an exhaustive search. It can be seen that the fraction of covered nodes is a non-decreasing function of the number of levels K, which corroborate our theoretical result presented in Lemma 4.4.1.

Next, we explore some properties of the fraction of covered nodes with parameter a for a *perfect a-ary* tree topology. Before discussing our result, we define the parameter a_{min} as follows. For a fixed K and attacker's cost budget $C_{budget}^{attacker}$, a_{min} is defined as the minimum value of a for which the attacker cannot blind the network or cover 50% or more nodes. So we can restrict our

analysis to $a_{min} \leq a \leq a_{max}$. Notice that, the attacker cannot blind all the trees $T(K, a)$ for which $a \geq a_{min}$ and can blind all the trees $T(K, a)$ for which $a < a_{min}$.

Lemma 4.4.2. *In a perfect a-ary tree topology, the fraction of covered nodes $\frac{\sum_{k=1}^{K} P_k B_k}{\sum_{k=1}^{K} N_k}$ by an attacker with cost budget $C_{budget}^{attacker}$ in an optimal attack is a decreasing function of parameter a for a perfect a-ary tree topology for $a \geq a_{min} \geq 2$.*

Proof. As before, let us denote the optimal attack configuration for a K level *perfect a-ary* tree topology $T(K,\ a)$ by $\{B_k^1\}_{k=1}^{K}$ and the optimal attack configuration for a *perfect (a+1)-ary* tree topology $T(K, a+1)$ by $\{B_k^2\}_{k=1}^{K}$ given the cost budget $C_{budget}^{attacker}$. To prove the lemma, it is sufficient to show that

$$\frac{\sum_{k=1}^{K} P_k^2 B_k^2}{\sum_{k=1}^{K} N_k^2} < \frac{\sum_{k=1}^{K} P_k^1 B_k^2}{\sum_{k=1}^{K} N_k^1} \leq \frac{\sum_{k=1}^{K} P_k^1 B_k^1}{\sum_{k=1}^{K} N_k^1} \tag{4.27}$$

where N_k^1 is the number of nodes at level k in $T(K, a)$, N_k^2 is the number of nodes at level k in $T(K, a+1)$, P_k^1 is the profit of attacking a node at level k in $T(K, a)$ and P_k^2 is the profit of attacking a node at level k in $T(K, a+1)$. Observe that, an interpretation of (4.27) is that the attacker is using the attack configuration $\{B_k^2\}_{k=1}^{K}$ to attack $T(K,\ a)$. However, one might suspect that the set $\{B_k^2\}_{k=1}^{k=K}$ is not a valid solution. More specifically, the set $\{B_k^2\}_{k=1}^{k=K}$ is not a valid solution in the following two cases:

1. $min(B_k^2,\ N_k^1) = N_k^1$ *for any k:* For example, if $N_1^1 = 4$ for $T(K, 4)$ and $B_1^2 = 5$ for $T(K, 5)$ then it will not be possible for the attacker to attack 5 nodes at level 1 in $T(K, 4)$ because the total number of nodes at level 1 is 4. In this case, $\{B_k^2\}_{k=1}^{K}$ might not be a valid attack configuration for the tree $T(K, a)$.

2. $\{B_k^2\}_{k=1}^{k=K}$ *is an overlapping set*[3] *for $T(K, a)$:* For example, for $T(2, 3)$ if $B_1^2 = 2$ and $B_2^2 = 4$,

[3]We call B_k and B_{k+x} are overlapping, if the summation of B_k^{k+x} and B_{k+x} is greater than N_{k+x}, where B_k^{k+x} is the number of nodes covered by the attack configuration B_k at level $k+x$. In a non-overlapping case, the attacker can always arrange nodes $\{B_k\}_{k=1}^{K}$ such that each path in the network has at most one Byzantine.

then, B_1^2 and B_2^2 are overlapping. In this case, $\{B_k^2\}_{k=1}^K$ might not be a valid attack configuration for the tree $T(K, a)$.

However, both of the above conditions imply that the attacker can blind the network with $C_{budget}^{attacker}$ (See Appendix A.6), which cannot be true for $a \geq a_{min}$, and, therefore, $\{B_k^2\}_{k=1}^K$ will indeed be a valid solution. Therefore, (4.27) is sufficient to prove the lemma.

Notice that, the second inequality in (4.27) follows due to the fact that $\{B_k^2\}_{k=1}^K$ may not be the optimal attack configuration for topology $T(K, a)$. To prove the first inequality in (4.27), we first consider the case where attack configuration $\{B_k^2\}_{k=1}^{k=K}$ contains only one node, i.e., $B_k^2 = 1$ for some k, and show that $\frac{P_k^2}{\sum_{k=1}^K N_k^2} < \frac{P_k^1}{\sum_{k=1}^K N_k^1}$. Substituting $P_k^1 = \dfrac{a^{K-k+1}-1}{a-1}$ for some k and $\sum_{k=1}^K N_k^1 = \dfrac{a(a^K-1)}{a-1}$ in the left side inequality of (4.27), we have

$$\frac{(a)^{K-k+1}-1}{(a)((a)^K-1)} > \frac{(a+1)^{K-k+1}-1}{(a+1)((a+1)^K-1)}.$$

After some simplification, the above condition becomes

$$\begin{aligned}&(a+1)^{K+1}[(a)^{K-k+1}-1] - (a)^{K+1}[(a+1)^{K-k+1}-1]\\ &+(a)[(a+1)^{K-k+1}-1] - (a+1)[(a)^{K-k+1}-1] > 0. \end{aligned} \tag{4.28}$$

In Appendix A.7, we show that

$$(a)[(a+1)^{K-k+1}-1] - (a+1)[(a)^{K-k+1}-1] > 0 \tag{4.29}$$

and

$$(a+1)^{K+1}[(a)^{K-k+1}-1] - (a)^{K+1}[(a+1)^{K-k+1}-1] \geq 0. \tag{4.30}$$

From (4.30) and (4.29), condition (4.28) holds.

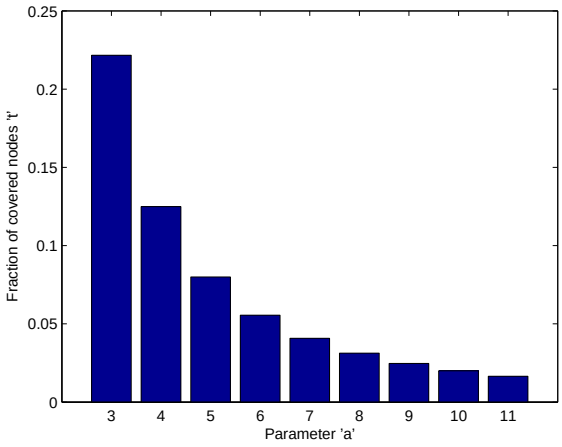


Fig. 4.4: Fraction of nodes covered vs Parameter a when $K = 6$, parameter a is varied from 3 to 11, $[c_1, \cdots, c_9] = [52,\ 48,\ 24,\ 16,\ 12,\ 8,\ 10,\ 6,\ 4]$, and $C_{budget}^{attacker} = 50$

Since we have proved that

$$\frac{P_k^2}{\sum_{k=1}^K N_k^2} < \frac{P_k^1}{\sum_{k=1}^K N_k^1} \text{ for all } 1 \leq k \leq K,$$

to generalize the proof for any arbitrary attack configuration $\{B_k^2\}_{k=1}^K$ we multiply both sides of the above inequality with B_k^2 and sum it over all $1 \leq k \leq K$ inequalities. Now, we have

$$\frac{\sum_{k=1}^K P_k^2 B_k^2}{\sum_{k=1}^K N_k^2} < \frac{\sum_{k=1}^K P_k^1 B_k^2}{\sum_{k=1}^K N_k^1}.$$

□

Next, to gain insights into the solution, we present some numerical results in Figure 4.4 that corroborate our theoretical results. We plot the fraction of covered nodes by the Byzantines as a function of the parameter a in the tree. We assume that the parameter $K = 6$ and vary a from 3 to 11. We also assume that the cost to attack nodes at different levels are given by $[c_1, \cdots, c_9] = [52,\ 48,\ 24,\ 16,\ 12,\ 8,\ 10,\ 6,\ 4]$ and the cost budget of the attacker is $C_{budget}^{attacker} = 50$. For each $T(6,\ a)$ we find the optimal attack configuration $\{B_k\}_{k=1}^K$ by an exhaustive search. It can be seen that the fraction of covered nodes is a decreasing function of the parameter a, which corroborate

our theoretical result presented in Lemma 4.4.2.

Note that, while deriving the results in Lemma 4.4.1 and Lemma 4.4.2 we have made no additional assumptions on how the attack configuration $\{B_k\}_{k=1}^{K}$ is obtained, so that the two lemmas would still hold even if the cost of deploying a node ($c_k^{network}$) is different from the cost of attacking ($c_k^{attacker}$) it. Further, as noted in the chapter, Lemma 4.4.1 and Lemma 4.4.2 suggest that the solution of the upper level problem, i.e., (K, a), is independent of the solution of lower level problem, i.e., $\{B_k\}_{k=1}^{K}$. In other words, even if $c_k^{network} \neq c_k^{attacker}$, the proposed solution approach would still hold.

Next, based on the above Lemmas we present an algorithm which can solve our robust perfect a-ary tree topology design problem (bi-level programming problem) efficiently.

4.4.2 Algorithm for solving Robust Perfect a-ary Tree Topology Design Problem

Based on Lemma 4.4.1 and Lemma 4.4.2, we present a polynomial time algorithm for solving the robust perfect a-ary tree topology design problem. Observe that, the robust network design problem is equivalent to designing perfect a-ary tree topology with minimum K and maximum a that satisfy network designer's constraints. In Algorithm 4.1, we start with the solution candidate $(K_{min},\ a_{max})$. First, the algorithm finds the largest integer $(a_{max} - l),\ l \geq 0$ that satisfies the cost expenditure constraint. If this value violates the hardware constraint, i.e., $(a_{max} - l) < a_{min}$, we will not have any feasible solution which satisfies the network designer's constraints. Next, the algorithm checks if $(K_{min},\ (a_{max} - l))$ satisfies the total number of nodes constraint. If it does, this will be the solution for the problem, otherwise, we increase K_{min} by one, i.e., $K_{min} \leftarrow K_{min} + 1$. Now, we have a new solution candidate $(K_{min} + 1,\ (a_{max} - l))$ and the algorithm solves the problem recursively in this manner.

This procedure greatly reduces the complexity because we do not need to solve the lower level problem in this case. Next, we prove that Algorithm 4.1 indeed yields an optimal solution.

Algorithm 4.1 Robust Perfect a-ary Tree Topology Design

Require: $c_k > c_{k+1} \qquad for\ k = 1, ..., K-1$

1: $K \leftarrow K_{min}$; $a \leftarrow a_{max}$
2: **if** $\left(\sum_{k=1}^{K} c_k a^k > C_{budget}^{network}\right)$ **then**
3: Find the largest integer $a - \ell$, $\ell \geq 0$, such that $\sum_{k=1}^{K} c_k (a-\ell)^k \leq C_{budget}^{network}$
4: **if** $(a - \ell < a_{min})$ **then**
5: **return** (ϕ, ϕ)
6: **else**
7: $a \leftarrow a - \ell$
8: **end if**
9: **end if**
10: **if** $\left(\sum_{k=1}^{K} a^k \geq N_{min}\right)$ **then**
11: **return** (K, a)
12: **else**
13: $K \leftarrow K + 1$
14: **return to** Step 2
15: **end if**

Lemma 4.4.3. *Robust Perfect a-ary Tree Topology Design algorithm (Algorithm 4.1) yields an optimal solution $(K^*,\ a^*)$, if one exists.*

Proof. Assume that the optimal solution exists. Let us denote by $(K^*,\ a^*)$, the optimal solution given by Algorithm 4.1. The main idea behind our proof is that any solution (K, a) with $K \geq K^*$ and $a \leq a^*$ cannot perform better than $(K^*,\ a^*)$ as suggested by Lemma 4.4.1 and Lemma 4.4.2. This property implies that the search should start with the smallest possible K and simultaneously the largest a, i.e., (K_{min}, a_{max}).

Notice that, our algorithm searches for the feasible solution with the smallest K and the largest a. Any feasible solution $(K,\ a)$ satisfies the following two conditions:

1. $\sum_{k=1}^{K} c_k a^k \leq C_{budget}^{network}$;
2. $\sum_{k=1}^{K} a^k \geq N_{min}$.

By Lemma 4.4.2, if (K, a) is a feasible solution, then (K, a') with $a' < a$ will not be a better solution than (K, a). Hence, for a given K, Step 3 only locates the solution with largest a for a given K. Furthermore, if both (K, a) and (K', a') satisfy Condition 1 and $K < K'$, then $a \geq a'$. Hence, for a given K, the largest a in the current iteration satisfying Condition 1 cannot be larger than the a found in the previous iteration. This verifies that $\ell \geq 0$ is a sufficient condition to find the largest a in Step 3.

Next, we prove that Algorithm 4.1 stops when the first feasible solution has been found. That is, the first feasible solution found by the algorithm is $(K^*,\ a^*)$. Let $(K^1,\ a^1)$ be the first feasible solution found by Algorithm 4.1. It can be observed from the algorithm that $K^* \geq K^1$ since the algorithm increases K from its smallest possible value and has not found a feasible solution until $K = K^1$. It is clear that the next feasible solution (K, a) must have $K > K^1$ and $a \leq a^1$, since, the algorithm increases K and it satisfies Condition 1. As suggested by Lemma 4.4.1 and Lemma 4.4.2, (K, a) cannot be a better solution than $(K^1,\ a^1)$. Hence, $K^* = K^1$ and $(K^1,\ a^1)$ is equal to $(K^*,\ a^*)$.

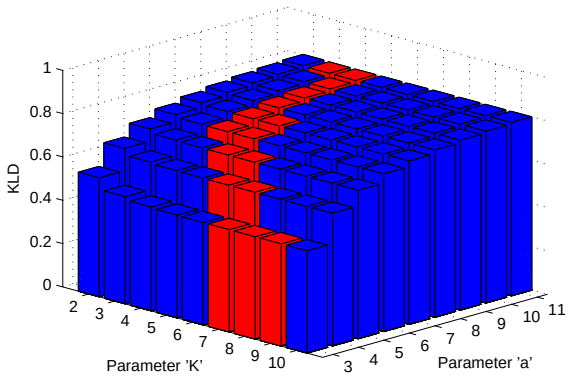


Fig. 4.5: KLD vs Parameters K and a when $(P_d, P_{fa}) = (0.8, 0.2)$, $C_{budget}^{network} = 400000$, $C_{budget}^{attacker} = 50$ and $N_{min} = 1400$

It can be seen that if there is no solution, then the algorithm will return $(\emptyset, \emptyset)$. This is due to the fact that if $a - \ell < a_{min}$, then no a can satisfy Condition 1 for current and further iterations. Hence, the algorithm terminates and returns $(\emptyset, \emptyset)$. □

Next, to gain insights into the solution, we present some numerical results in Figure 4.5 that corroborate our theoretical results. We plot the $\min_{P_{1,0}, P_{0,1}}$ KLD for all the combinations of parameter K and a in the tree. We vary the parameter K from 2 to 10 and a from 3 to 11. We also assume that the costs to attack nodes at different levels are given by $[c_1, \cdots, c_{10}] = [52,\ 50,\ 25,\ 24,\ 16,\ 10,\ 8,\ 6,\ 5,\ 4]$, and cost budgets of the network and the attacker are given by $C_{budget}^{network} = 400000$, $C_{budget}^{attacker} = 50$, respectively. The node budget constraint is assumed to be $N_{min} = 1400$. For each $T(K,\ a)$, we find the optimal attack configuration $\{B_k\}_{k=1}^{K}$ by an exhaustive search. All the feasible solutions are plotted in red and unfeasible solutions are plotted in blue. Notice that, $T(K_{min},\ a_{max})$ which is $T(2,\ 11)$ is not a feasible solution and, therefore, if we use Algorithm 4.1 it will try to find the feasible solution which has minimum possible deviation from $T(K_{min},\ a_{max})$. It can be seen that the optimal solution $T(3,\ 11)$ has minimum possible deviation from the $T(K_{min},\ a_{max})$, which corroborate our algorithm.

4.5 Discussion

In this chapter, we have considered distributed detection in perfect a-ary tree topologies in the presence of Byzantines, and characterized the power of attack analytically. We provided closed-form expressions for minimum attacking power required by the Byzantines to blind the FC. We obtained closed form expressions for the optimal attacking strategies that minimize the detection error exponent at the FC. We also looked at the possible counter-measures from the FC's perspective to protect the network from these Byzantines. We formulated the robust topology design problem as a bi-level program and provided an efficient algorithm to solve it.

CHAPTER 5
DISTRIBUTED DETECTION WITH LABELED BYZANTINE DATA: TREE TOPOLOGY

5.1 Introduction

In the previous chapter, we studied the problem of distributed detection in perfect tree networks (all intermediate nodes in the tree have the same number of children) with Byzantines under the assumption that the FC does not know which decision bit is sent from which node and assumes each received bit to originate from nodes at depth k with a certain probability. Under this assumption, the attacker's aim was to maximize the false alarm probability for a fixed detection probability. When the number of nodes is large, by Stein's lemma [23], we know that the error exponent of the false alarm probability can be used as a surrogate for the false alarm probability. Thus, the optimal attacking strategy was obtained by making the error exponent of the false alarm probability at the FC equal to zero, which makes the decision fusion scheme completely incapable (blind). Some counter-measures were also proposed to protect the network from such Byzantines.

In contrast to the previous chapter, in this chapter, the problem of distributed detection in regular tree networks[1] with Byzantines is addressed in a more practical setup where the FC has the

[1]For a regular tree, intermediate nodes at different levels are allowed to have different degrees, i.e., number of children.

knowledge of which bit is transmitted from which node. Note that, in practice, the FC knows which bit is transmitted from which node, e.g., using MAC schemes[2], and can utilize this information to improve system performance. Next, for the analysis of the optimal attack, we consider nodes residing at different levels of the tree to have different detection performance. We also allow Byzantines residing at different levels of the tree to have different attacking strategies and, therefore, provide a more general and comprehensive analysis of the problem as compared to the previous chapter. We also study the problem from the network designer's perspective. We model the strategic interaction between the FC and the attacker as a Leader-Follower (Stackelberg) game and identify attacker and defender (FC) equilibrium strategies. The knowledge of these equilibrium strategies can later be used to implement the optimal detector at the FC. Based on the information regarding which bit is transmitted from which node, we propose schemes to mitigate the effect of the Byzantines. More specifically, we propose a simple yet efficient reputation based scheme, which works even if the FC is blinded, to identify Byzantines in tree networks and analytically evaluate its performance.

The rest of the chapter is organized as follows. Section 5.2 introduces the system model. In Section 5.3, we study the problem from Byzantine's perspective and provide closed form expressions for optimal attacking strategies. In Section 5.4, we investigate the problem of designing optimal distributed detection parameters in the presence of Byzantines. In Section 5.5, we model the strategic interaction between the FC and the attacker as a Leader-Follower (Stackelberg) game and find equilibrium strategies. In Section 5.6, we introduce an efficient Byzantine identification scheme and analyze its performance. Finally, Section 5.7 concludes the chapter.

5.2 System Model

We consider a distributed detection system organized as a regular tree network rooted at the FC. For a regular tree, all the leaf nodes are at the same level (or depth) and all the intermediate nodes at level k have degree a_k. The regular tree is assumed to have a set $\mathcal{N} = \{\mathbb{N}_k\}_{k=1}^{K}$ of transceiver

[2]In practice, one possible way to achieve this is by using the buffer-less TDMA MAC protocol, in which, distinct non-overlapping time slots are assigned (scheduled) to the nodes for communication.

nodes, where $|\mathbb{N}_k| = N_k$ is the total number of nodes at level k. We assume that the depth of the tree is $K > 1$ and $a_k \geq 2$. The total number of nodes in the network is denoted as $N = \sum_{k=1}^{K} N_k$ and $\mathcal{B} = \{\mathbb{B}_k\}_{k=1}^{K}$ denotes the set of Byzantine nodes with $|\mathbb{B}_k| = B_k$, where $\mathbb{B}_k$ is the set of Byzantines at level k. The set containing the number of Byzantines residing at each level k, $1 \leq k \leq K$, is referred to as an attack configuration, i.e., $\{B_k\}_{k=1}^{K} = \{|\mathbb{B}_k|\}_{k=1}^{K}$.

We consider a binary hypothesis testing problem with two hypotheses H_0 (signal is absent) and H_1 (signal is present). Under each hypothesis, it is assumed that the observations $Y_{k,i}$ at each node i at level k are conditionally independent. Each node i at level k acts as a source in the sense that it makes a one-bit (binary) local decision $v_{k,i} \in \{0, 1\}$ regarding the absence or presence of the signal using the likelihood ratio test (LRT). We denote the probabilities of detection and false alarm of a node at level k by $P_d^k = P(v_{k,i} = 1|H_1)$ and $P_{fa}^k = P(v_{k,i} = 1|H_0)$, respectively, which are functions of λ_k and hold for both Byzantines and honest nodes. After making its one-bit local decision $v_{k,i} \in \{0, 1\}$, node i at level k sends $u_{k,i}$ to its parent node at level $k-1$, where $u_{k,i} = v_{k,i}$ if i is an honest node, but for a Byzantine node i, $u_{k,i}$ need not be equal to $v_{k,i}$. Node i at level k also receives the decisions $u_{k',j}$ of all successors j at levels $k' \in [k+1, K]$, which are forwarded to node i by its immediate children, and forwards them to its parent node at level $k - 1$. We assume error-free communication between children and the parent nodes.

Next, we present a mathematical model for the Byzantine attack as defined in 2.2.3. We denote the strategies $P_{j,1}^H(k)$, $P_{j,0}^H(k)$ and $P_{j,1}^B(k)$, $P_{j,0}^B(k)$ ($j \in \{0, 1\}$ and $k = 1, \cdots, K$) for the honest and Byzantine nodes at level k, respectively, where $P_k(x = a|y = b)$ is the conditional probability that a node at level k sends a to its parent when it receives b from its child or its actual decision is b. For notational convenience, we use $(P_{1,0}^k, P_{0,1}^k)$ to denote the flipping probability of the Byzantine node at level k. Furthermore, we assume that if a node (at any level) is a Byzantine, then none of its ancestors and successors are Byzantine (non-overlapping attack configuration); otherwise, the effect of a Byzantine due to other Byzantines on the same path may be nullified (e.g., Byzantine ancestor re-flipping the already flipped decisions of its successors). This means that every path from a leaf node to the FC will have at most one Byzantine. Notice that, for the attack configuration

$\{B_k\}_{k=1}^K$, the total number of corrupted paths (i.e., paths containing a Byzantine node) from level k to the FC are $\sum_{i=1}^k B_i \frac{N_k}{N_i}$, where $B_i \frac{N_k}{N_i}$ is the total number of nodes covered[3] at level k by the presence of B_i Byzantines at level i. If we denote $\alpha_k = \frac{B_k}{N_k}$, then, $\frac{\sum_{i=1}^k B_i \frac{N_k}{N_i}}{N_k} = \sum_{i=1}^k \alpha_i$ is the fraction of decisions coming from level k that encounter a Byzantine along the way to the FC. For a large network, due to the law of large numbers, one can approximate the probability that the FC receives the flipped decision $\bar{x}$ of a given node at level k when its actual decision is x as $\beta_{\bar{x},x}^k = \sum_{j=1}^k \alpha_j P_{\bar{x},x}^j,\ x \in \{0,1\}$.

We consider the distributed detection problem under the Neyman-Pearson (NP) criterion. The FC receives decision vectors, $[\mathbf{z_1}, \cdots, \mathbf{z_K}]$, where $\mathbf{z_k}$ for $k \in \{1, \cdots, K\}$ is a decision vector with its elements being $z_1, \cdots, z_{N_k}$, from the nodes at different levels of the tree. Then the FC makes the global decision about the phenomenon by employing the LRT. Due to system vulnerabilities, some of the nodes may be captured by the attacker and reprogrammed to transmit false information to the FC to degrade detection performance. We assume that the only information available at the FC is the probability $\beta_{\bar{x},x}^k$, which is the probability with which the data coming from level k has been falsified. Using this information, the FC calculates the probabilities $\pi_{j,0}^k = P(z_i = j|H_0, k)$ and $\pi_{j,1}^k = P(z_i = j|H_1, k)$, which are the distributions of received decisions z_i originating from level k and arriving to the FC under hypotheses H_0 and H_1. The FC makes its decision regarding the absence or presence of the signal using the following likelihood ratio test

$$\prod_{k=1}^{K} \left(\frac{\pi_{1,1}^k}{\pi_{1,0}^k}\right)^{s_k} \left(\frac{1-\pi_{1,1}^k}{1-\pi_{1,0}^k}\right)^{N_k - s_k} \underset{H_0}{\overset{H_1}{\gtrless}} \eta \tag{5.1}$$

where s_k is the number of decisions that are equal to one and originated from level k, and the threshold η is chosen in order to minimize the missed detection probability (P_M) while keeping the false alarm probability (P_F) below a fixed value δ.[4]

[3]Node i at level k' covers (or can alter the decisions of) all its children at levels $k'+1$ to K and itself. In other words, the total number of covered nodes is equivalent to the total number of corrupted paths (i.e., paths containing a Byzantine node) in the network.

[4]This type of problem setup is important, for instance, in Cognitive Radio Networks (CRN). In order to coexist with the primary user (PU), secondary users (SUs) must guarantee that their transmissions will not interfere with the transmission of the PU who have higher priority to access the spectrum.

Next, we derive a closed form expression for the optimal missed detection error exponent for tree networks in the presence of Byzantines, which will later be used as a surrogate for the probability of missed detection.

Proposition 5.2.1. *For a K level tree network employing the detection scheme as given in* (5.1)*, the asymptotic detection performance (i.e., $N_1 \to \infty$) can be characterized using the missed detection error exponent given below*

$$D = \sum_{k=1}^{K} N_k \left[\sum_{j \in \{0,1\}} \pi_{j,0}^k \log \frac{\pi_{j,0}^k}{\pi_{j,1}^k} \right]. \tag{5.2}$$

Proof. Let $\mathbf{Z} = [\mathbf{Z}_1, \cdots, \mathbf{Z}_{N_1}]$ denote the received decision vectors from the nodes at level 1, where $\mathbf{Z_i}$ is the decision vector forwarded by the node i at level 1 to the FC. Observe that, $\mathbf{Z}_i$ for $i = 1$ to N_1 are independent and identically distributed (i.i.d.). Therefore, using Stein's lemma [23], when $N_1 \to \infty$, the optimal error exponent for the detection scheme as given in (5.1) is the Kullback-Leibler divergence (KLD) [56] between the distributions $P(\mathbf{Z}|H_0)$ and $P(\mathbf{Z}|H_1)$. The summation term in (5.2) follows from the additive property of the KLD for independent distributions. □

Note that, (5.2) can be compactly written as $\sum_{k=1}^{K} N_k D_k(\pi_{j,1}^k || \pi_{j,0}^k)$ with $D_k(\pi_{j,1}^k || \pi_{j,0}^k)$ being the KLD between the data coming from node i at level k under H_0 and H_1. The FC wants to maximize the detection performance, while, the Byzantine attacker wants to degrade the detection performance as much as possible which can be achieved by maximizing and minimizing the KLD, respectively. Next, we explore the optimal attacking strategies for the Byzantines that degrade the detection performance most by minimizing the KLD.

5.3 Optimal Byzantine Attack

Since the KLD is always non-negative, Byzantines attempt to choose $P(z_i = j|H_0, k)$ and $P(z_i = j|H_1, k)$ such that $D_k = 0,\ \forall k$. This is possible when

$$P(z_i = j|H_0, k) = P(z_i = j|H_1, k) \qquad \forall j \in \{0, 1\},\ \forall k. \tag{5.3}$$

Notice that, $\pi^k_{j,0} = P(z_i = j|H_0, k)$ and $\pi^k_{j,1} = P(z_i = j|H_1, k)$ can be expressed as

$$\pi^k_{1,0} = \beta^k_{1,0}(1 - P^k_{fa}) + (1 - \beta^k_{0,1})P^k_{fa} \tag{5.4}$$

$$\pi^k_{1,1} = \beta^k_{1,0}(1 - P^k_{d}) + (1 - \beta^k_{0,1})P^k_{d}. \tag{5.5}$$

with $\beta^k_{1,0} = \sum_{j=1}^k \alpha_j P^j_{1,0}$ and $\beta^k_{0,1} = \sum_{j=1}^k \alpha_j P^j_{0,1}$. Substituting (5.4) and (5.5) in (5.3) and after simplification, the condition to make the $D = 0$ for a K-level network becomes $\sum_{j=1}^k \alpha_j (P^j_{1,0} + P^j_{0,1}) = 1,\ \forall k$.

Notice that, when $\sum_{j=1}^k \alpha_j < 0.5$, there does not exist any attacking probability distribution $(P^j_{0,1}, P^j_{1,0})$ that can make $D_k = 0$, and, therefore, the KLD cannot be made zero. In the case of $\sum_{j=1}^k \alpha_j = 0.5$, there exists a unique solution $(P^j_{0,0}, P^j_{1,0}) = (1, 1),\ \forall j$ that can make $D_k = 0$, $\forall k$. For the $\sum_{j=1}^k \alpha_j > 0.5$ case, there exist infinitely many attacking probability distributions $(P^j_{0,1}, P^j_{1,0})$ which can make $D_k = 0, \forall k$. Thus, we have the following result.

Lemma 5.3.1. *In a tree network with K levels, the minimum number of Byzantines needed to make the Kullback-Leibler divergence (KLD) between the distributions $P(\mathbf{Z}|H_0)$ and $P(\mathbf{Z}|H_1)$ equal to zero (or to make $D_k = 0,\ \forall k$) is given by $B_1 = \left\lceil \frac{N_1}{2} \right\rceil$.*

Proof. The proof follows from the fact that the condition $\sum_{j=1}^k \alpha_j = 0.5,\ \forall k$, is equivalent to $\alpha_1 = 0.5,\ \alpha_k = 0,\ \forall k = 2, \cdots, K$. □

Next, we explore the optimal attacking probability distribution $(P^k_{0,1}, P^k_{1,0})$ that minimizes D_k when $\sum_{j=1}^k \alpha_j < 0.5$, i.e., in the case where the attacker cannot make $D = 0$. To analyze the problem, first we investigate the properties of D_k with respect to $(P^k_{0,1}, P^k_{1,0})$ assuming

$(P^j_{0,1}, P^j_{1,0}),\ 1 \leq j \leq k-1$ to be fixed. We show that attacking with symmetric flipping probabilities is the optimal strategy in the region where the attacker cannot make $D_k = 0$. In other words, attacking with $P^k_{1,0} = P^k_{0,1}$ is the optimal strategy for the Byzantines.

Lemma 5.3.2. *In the region where the attacker cannot make $D_k = 0$, i.e., for $\sum_{j=1}^{k} \alpha_j < 0.5$, the optimal attacking strategy comprises of symmetric flipping probabilities $(P^k_{0,1} = P^k_{1,0} = p)$. In other words, any non zero deviation $\epsilon_i \in (0, p]$ in flipping probabilities $(P^k_{0,1}, P^k_{1,0}) = (p-\epsilon_1, p-\epsilon_2)$, where $\epsilon_1 \neq \epsilon_2$, will result in an increase in D_k.*

Proof. Please see Appendix A.8. □

In the next theorem, we present the solution for the optimal attacking probability distribution $(P^k_{j,1}, P^k_{j,0})$ that minimizes D_k in the region where the attacker cannot make $D_k = 0$.

Theorem 5.3.3. *In the region where the attacker cannot make $D_k = 0$, i.e., for $\sum_{j=1}^{k} \alpha_j < 0.5$, the optimal attacking strategy is given by $(P^k_{0,1}, P^k_{1,0}) = (1, 1)$.*

Proof. Observe that, in the region where the attacker cannot make $D_k = 0$, the optimal strategy comprises of symmetric flipping probabilities $(P^k_{0,1} = P^k_{1,0} = p)$. The proof is complete if we show that D_k is a monotonically decreasing function of the flipping probability p.

After plugging in $(P^k_{0,1}, P^k_{1,0}) = (p, p)$ in (5.4) and (5.5), we get

$$\pi^k_{1,1} = [\beta^{k-1}_{1,0}(1 - P^k_d) + (1 - \beta^{k-1}_{0,1})P^k_d] + [\alpha_k(p - P^k_d(2p)) + P^k_d] \tag{5.6}$$

$$\pi^k_{1,0} = [\beta^{k-1}_{1,0}(1 - P^k_{fa}) + (1 - \beta^{k-1}_{0,1})P^k_{fa}] + [\alpha_k(p - P^k_{fa}(2p)) + P^k_{fa}]. \tag{5.7}$$

Now we show that D_k is a monotonically decreasing function of the parameter p or in other words, $\frac{dD_k}{dp} < 0$. After plugging in $\pi^{k'}_{1,1} = \alpha_k(1 - 2P^k_d)$ and $\pi^{k'}_{1,0} = \alpha_k(1 - 2P^k_{fa})$ in the expression of $\frac{dD_k}{dp}$ and rearranging the terms, the condition $\frac{dD_k}{dp} < 0$ becomes

$$(1 - 2P^k_d)\left(\frac{1 - \pi^k_{1,0}}{1 - \pi^k_{1,1}} - \frac{\pi^k_{1,0}}{\pi^k_{1,1}}\right) + (1 - 2P^k_{fa}) \log\left(\frac{1 - \pi^k_{1,1}}{1 - \pi^k_{1,0}} \frac{\pi^k_{1,0}}{\pi^k_{1,1}}\right) < 0 \tag{5.8}$$

Since $P_d^k > P_{fa}^k$ and $\beta_{\bar{x},x}^k < 0.5$, we have $\pi_{1,1}^k > \pi_{1,0}^k$. Now, using the fact that $\frac{1-P_d^k}{1-P_{fa}^k} > \frac{1-2P_d^k}{1-2P_{fa}^k}$ and (A.51), we have

$$\frac{1-2P_d^k}{1-2P_{fa}^k}\left[\frac{1-\pi_{1,0}^k}{1-\pi_{1,1}^k}-\frac{\pi_{1,0}^k}{\pi_{1,1}^k}\right] < (\pi_{1,1}^k-\pi_{1,0}^k)\left[\frac{1}{\pi_{1,1}^k}+\frac{1}{1-\pi_{1,0}^k}\right] \tag{5.9}$$

$$\Leftrightarrow \quad \frac{1-2P_d^k}{1-2P_{fa}^k}\left[\frac{1-\pi_{1,0}^k}{1-\pi_{1,1}^k}-\frac{\pi_{1,0}^k}{\pi_{1,1}^k}\right]+\left[\frac{\pi_{1,0}^k}{\pi_{1,1}^k}-1\right] < 1-\frac{1-\pi_{1,1}^k}{1-\pi_{1,0}^k}. \tag{5.10}$$

Applying the logarithm inequality $(x-1) \geq \log x \geq \frac{x-1}{x}$, for $x > 0$ to (5.10), one can prove that (5.8) is true. □

Next, to gain insights into the solution, we present some numerical results in Figure 5.1. We plot D_k as a function of the flipping probabilities $(P_{1,0}^k,\ P_{0,1}^k)$. We assume that the probability of detection is $P_d^k = 0.8$, the probability of false alarm is $P_{fa}^k = 0.2$, and the probability that the bit coming from level k encounters a Byzantine is $\sum_{j=1}^k \alpha_j = 0.4$. We also assume that $P_{0,1}^k = P_{0,1}$ and $P_{1,0}^k = P_{1,0}, \forall k$. It can be seen that the optimal attacking strategy comprises of symmetric flipping probabilities and is given by $(P_{0,1}^k, P_{1,0}^k) = (1,1)$, which corroborates our theoretical result presented in Lemma 5.3.2 and Theorem 5.3.3.

We have shown that, for all k,

$$D_k(P_{0,1}^k, P_{1,0}^k) \geq D_k(1,1). \tag{5.11}$$

Now, by multiplying both sides of (5.11) by N_k and summing it over all K we can show that the KLD, D, is minimized by $(P_{0,1}^k, P_{1,0}^k) = (1,1)$, for all k, in the region $\sum_{k=1}^{K} \alpha_k < 0.5$.

Now, we explore some properties of D_k with respect to $\sum_{j=1}^k \alpha_j$ in the region where the attacker cannot make $D_k = 0$, i.e., for $\sum_{j=1}^k \alpha_j < 0.5$. This analysis will later be used in exploring the problem from the network designer's perspective.

Lemma 5.3.4. *$D_k^* = \min_{(P_{j,1}^k, P_{j,0}^k)} D_k(\pi_{j,1}^k || \pi_{j,0}^k)$ is a continuous, decreasing and convex function of*

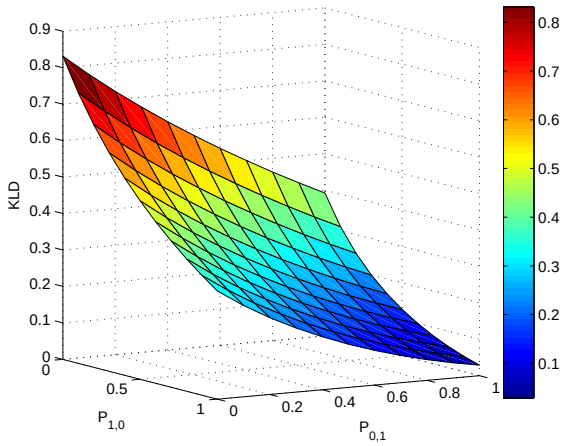


Fig. 5.1: KLD D_k vs. flipping probabilities when $P_d^k = 0.8$, $P_{fa}^k = 0.2$, and the probability that the bit coming from level k encounters a Byzantine is $\sum_{j=1}^{k} \alpha_j = 0.4$.

$\sum_{j=1}^{k} \alpha_j$ *for* $\sum_{j=1}^{k} \alpha_j < 0.5$.

Proof. The continuity of $D_k(\pi_{j,1}^k || \pi_{j,0}^k)$ with respect to the involved distributions implies the continuity of D_k^*. To show that D_k^* is a decreasing function of $t = \sum_{j=1}^{k} \alpha_j$, we use the fact that $\arg\min_{(P_{0,1}^k, P_{1,0}^k)} D_k(\pi_{j,1}^k || \pi_{j,0}^k)$ is equal to $(1,1)$ for $\sum_{j=1}^{k} \alpha_j < 0.5$ (as shown in Theorem 5.3.3). After plugging $(P_{0,1}^k, P_{1,0}^k) = (1,1),\ \forall k$, in the KLD expression, it can be shown that $\frac{dD_k}{dt} < 0$. Hence, D_k^* is a monotonically decreasing function of $\sum_{j=1}^{k} \alpha_j$ for $\sum_{j=1}^{k} \alpha_j < 0.5$. The convexity of D_k^* follows from the fact that $D_k^*(\pi_{j,1}^k || \pi_{j,0}^k)$ is convex in $\pi_{j,1}^k$ and $\pi_{j,0}^k$, which are affine transformations of $\sum_{j=1}^{k} \alpha_j$ (Note that, convexity holds under affine transformation). □

It is worth noting that Lemma 5.3.4 suggests that minimization/maximization of $\sum_{j=1}^{k} \alpha_j$ is equivalent to minimization/maximization of D_k. Using this fact, one can consider the probability that the bit coming from level k encounters a Byzantine (i.e., $t = \sum_{j=1}^{k} \alpha_j$) in lieu of D_k for optimizing the system performance.

Next, to gain insights into the solution, we present some numerical results in Figure 5.2. We plot $\min_{(P_{j,1}^k, P_{j,0}^k)} D_k$ as a function of the probability that the bit coming from level k encounters a Byzantine, i.e., t. We assume that the probabilities of detection and false alarm are $P_d^k = 0.8$ and $P_{fa}^k = 0.2$, respectively. Notice that, when $t = 0.5$, D_k between the two probability distributions

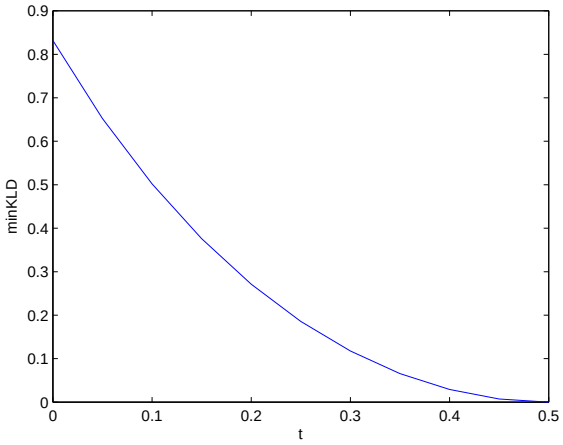


Fig. 5.2: $\min_{(P^k_{j,1},P^k_{j,0})} D_k$ vs probability that the bit coming from level k encounters a Byzantine for $P^k_d = 0.8$ and $P^k_{fa} = 0.2$.

becomes zero. It is seen that D^*_k is a continuous, decreasing and convex function of the fraction of covered nodes, t, for $t < 0.5$, which corroborates our theoretical result presented in Lemma 5.3.4.

Until now, we have explored the problem from the attacker's perspective. In the rest of the chapter, we look into the problem from a network designer's perspective and propose techniques to mitigate the effect of Byzantines. First, we study the problem of designing optimal distributed detection parameters in a tree network in the presence of Byzantines.

5.4 System Design in the Presence of Byzantines

For a fixed attack configuration $\{B_k\}_{k=1}^K$, the detection performance at the FC is a function of the local detectors used at the nodes in the tree network and the global detector used at the FC. This motivates us to study the problem of designing detectors, both at the nodes at different levels in a tree and at the FC, such that the detection performance is maximized. More specifically, we are interested in answering the question: How does the knowledge of the attack configuration $\{B_k\}_{k=1}^K$ affect the design of optimal distributed detection parameters?

By Stein's lemma [23], we know that in the NP setup for a fixed false alarm probability, the missed detection probability of the optimal detector can be minimized by maximizing the KLD.

For an optimal detector at the FC, the problem of designing the local detectors can be formalized as follows:

$$\max_{\{P_d^k, P_{fa}^k\}_{k=1}^K} \quad \sum_{k=1}^{K} N_k \sum_{j\in\{0,1\}} P(z_i = j|H_0, k) \log \frac{P(z_i = j|H_0, k)}{P(z_i = j|H_1, k)}. \tag{5.12}$$

The local detector design problem as given in (5.12) is a non-linear optimization problem. Furthermore, it is difficult to obtain a closed form solution for this problem. Next, we show that likelihood ratio tests remain optimal (under the conditional independence assumption) even in the presence of Byzantines and optimal decision rule for each node is independent of Byzantines' parameters.[5] To solve the problem, we need to find the pairs $\{P_d^k, P_{fa}^k\}_{k=1}^K$ which maximize the objective function as given in (5.12). However, P_d^k and P_{fa}^k are coupled and, therefore, cannot be optimized independently. Thus, we first analyze the problem of maximizing the KLD for a fixed P_{fa}^k. We assume that $P_{fa}^k = y_k$ and $P_d^k = y_k + x_k$. Next, we analyze the properties of KLD with respect to x_k, i.e., $(P_d^k - P_{fa}^k)$ in the region where attacker cannot blind the FC, i.e., for $\sum_{j=1}^k \alpha_j < 0.5$, in order to study the local detector design problem. Notice that, in the region $\sum_{j=1}^k \alpha_j \geq 0.5$, $D_k = 0$ and optimizing over local detectors does not improve the performance.

Lemma 5.4.1. *For a fixed $P_{fa}^k = y_k$, when $\sum_{j=1}^k \alpha_j < 0.5$, the KLD, D, as given in* (5.2) *is a monotonically increasing function of $x_k = (P_d^k - P_{fa}^k)$.*

Proof. To prove this, we calculate the partial derivative of D with respect to x_k. By substituting $P_{fa}^k = y_k$ and $P_d^k = y_k + x_k$ into (5.2), the partial derivative of D with respect to x_k can be calculated as

[5] In other words, under the assumption of conditional independence, an optimal decision rule for each node takes the form of a likelihood ratio test (LRT), with a suitably chosen threshold. In turn, optimization over the set of all thresholds can yield the desired solution.

$$\frac{\partial D}{\partial x_k} = N_k \frac{\partial}{\partial x_k}\left[\pi^k_{1,0}\log\frac{\pi^k_{1,0}}{\pi^k_{1,1}} + (1-\pi^k_{1,0})\log\frac{1-\pi^k_{1,0}}{1-\pi^k_{1,1}}\right]$$
$$\Leftrightarrow \quad \frac{\partial D}{\partial x_k} = N_k \pi^{k'}_{1,1}\left(\frac{1-\pi^k_{1,0}}{1-\pi^k_{1,1}} - \frac{\pi^k_{1,0}}{\pi^k_{1,1}}\right),$$

where $\pi^k_{1,0}$ and $\pi^k_{1,1}$ are as given in (5.4) and (5.5), respectively and $\pi^{k'}_{1,1} = (1-\beta^k_{0,1}-\beta^k_{1,0})$. Notice that,

$$\left(\frac{1-\pi^k_{1,0}}{1-\pi^k_{1,1}} - \frac{\pi^k_{1,0}}{\pi^k_{1,1}}\right) > 0$$
$$\Leftrightarrow \quad \pi^k_{1,1} > \pi^k_{1,0}.$$

Thus, the condition to make $\frac{\partial D}{\partial x_k} > 0$ simplifies to

$$\pi^{k'}_{1,1} > 0 \Leftrightarrow 1 > (\beta^k_{0,1} + \beta^k_{1,0}) \tag{5.13}$$

Substituting the values of $\beta^k_{1,0}$ and $\beta^k_{1,1}$, the above condition can be written as:

$$\sum_{j=1}^{k} \alpha_j P^j_{1,0} + \sum_{j=1}^{k} \alpha_j P^j_{0,1} < 1 \tag{5.14}$$
$$\Leftrightarrow \quad \sum_{j=1}^{k} \alpha_j (P^j_{1,0} + P^j_{0,1}) < 1 \tag{5.15}$$

The above condition is true for any $0 \leq P^j_{0,1}, P^j_{1,0} \leq 1$ when $\sum_{j=1}^{k} \alpha_j < 0.5$. This completes the proof. □

Lemma 5.4.1 suggests that one possible solution to maximize D is to choose the largest possible x_k constrained to $0 \leq x_k \leq 1 - y_k$. The upper bound results from the fact that $\{P^k_d, P^k_{fa}\}_{k=1}^{K}$ are probabilities and, thus, must be between zero and one. In other words, the solution is to maximize the probability of detection for a fixed value of probability of false alarm. In detection theory,

it is well known that the likelihood ratio based test is optimum for this criterion. Thus, under the conditional independence assumption, the likelihood ratio based test as given in (5.1) is optimal for local nodes, even in the presence of Byzantines, and the optimal operating points $\{P_d^{k*}, P_{fa}^{k*}\}_{k=1}^K$ are independent of the Byzantines' parameters $\{\alpha_k\}_{k=1}^K$.

The above result has the following important consequences: $1)$ search space is reduced from any arbitrary detector to likelihood ratio based detectors, $2)$ the threshold in the LRT can be optimized without any prior knowledge about the Byzantines' parameters $\{\alpha_k\}_{k=1}^K$. We further explore the problem from the network designer's (FC) perspective. In our previous analysis, we have assumed that the attack configuration $\{B_k\}_{k=1}^K$ is known and shown that the optimal local detector is independent of $\{\alpha_k\}_{k=1}^K$. However, notice that the KLD is the exponential decay rate of the error probability of the optimal detector. In other words, while optimizing over KLD, we implicitly assumed that the optimal detector, which is a likelihood ratio based detector, is used at the FC. Taking logarithm on both sides of (5.1), the optimal decision rule simplifies to

$$\sum_{k=1}^{K}[a_1^k s_k + a_0^k(N_k - s_k)] \underset{H_0}{\overset{H_1}{\gtrless}} \log\eta \tag{5.16}$$

where the optimal weights are given by $a_1^k = \log\frac{\pi_{1,1}^k}{\pi_{1,0}^k}$ and $a_0^k = \log\frac{1-\pi_{1,1}^k}{1-\pi_{1,0}^k}$. To implement the optimal detector, the FC needs to know the optimal weights a_j^k, which are functions of $\{\alpha_k\}_{k=1}^K$. In the next section, we are interested in answering the question: Is it possible for the FC to predict the attack configuration $\{B_k\}_{k=1}^K$ in the tree? The knowledge of this attack configuration can be used for determining the optimal detector at the FC to improve the system performance. Notice that, learning/estimation based techniques can be used on data to determine the attack configuration. However, the FC has to acquire a large amount of data coming from the nodes over a long period of time to accurately estimate $\{B_k\}_{k=1}^K$.

In the next section, we propose a novel technique to predict the attack configuration by considering the following scenario: The FC, acting first, commits to a defensive strategy by deploying the defensive resources to protect the tree network, while the attacker chooses its best response or

attack configuration after surveillance of this defensive strategy. Both, the FC and the Byzantines have to incur a cost to deploy the defensive resources and attack the nodes in the tree network, respectively. We consider both the FC and the attacker to be strategic in nature and model the strategic interaction between them as a Leader-Follower (Stackelberg) game. This formulation provides a framework for identifying attacker and defender (FC) equilibrium strategies, which can be used to implement the optimal detector. The main advantage of this technique is that the equilibrium strategies can be determined *a priori* and, therefore, there is no need to observe a large amount of data coming from the nodes over a long period of time to accurately estimate $\{B_k\}_{k=1}^{K}$.

5.5 Stackelberg Game for Attack Configuration Prediction Problems

We model the strategic interaction between the FC and the attacker as a Leader-Follower (Stackelberg) game. We assume that the FC has to incur a cost for deploying the network and the Byzantine has to incur a cost[6] for attacking the network. It is assumed that the network designer or the FC has a cost budget $C_{budget}^{network}$ and the attacker has a cost budget $C_{budget}^{attacker}$[7]. More specifically, the FC wants to allocate the best subset of defensive resources (denoted as $\{\tilde{c}_k\}_{k=1}^{K}$)[8] from a set of available defensive resources $\mathbb{C} = (c_1, \cdots, c_n)$ (arranged in a descending order, i.e., $c_1 \geq c_2 \cdots \geq c_n$), where $n \geq K$, complying with its budget constraint $C_{budget}^{network}$ to different levels of the tree network. After the FC allocates the defensive resources or budget to different levels of the tree network, an attacker chooses an attack configuration, $\{B_k\}_{k=1}^{K}$ complying with his budget constraint $C_{budget}^{attacker}$

[6]Due to variations in hardware complexity and the level of tamper-resistance present in nodes residing at different levels of the tree, the resources required to capture and tamper nodes at different levels may be different and, therefore, nodes have varying costs of being attacked.

[7]In this chapter, we assume that the attacker budget $C_{budget}^{attacker}$ is such that $\sum_{k=1}^{K} \alpha_k < 0.5$, i.e., the attacker cannot make $D_k = 0,\ \forall k$. Notice that, if the attacker can make $D_k = 0$ for some $k = l$, then, it can also make $D_k = 0,\ \forall k \geq l$. Also, $D_k = 0$ implies that $\pi_{1,1}^{k} = \pi_{1,0}^{k}$ and, therefore, the weights (a_1^k, a_0^k) in (5.16) are zero. In other words, the best the FC can do in the case when $D_k = 0,\ \forall k \geq l$ is to ignore or discard the decisions of the nodes residing at level $k \geq l$. This scenario is equivalent to using the tree network with $(l-1)$ levels for distributed detection.

[8]Let $\tilde{c}_k$ denote the resources deployed or budget allocated by the FC to protect or deploy a node at level k.

to maximally degrade the performance of the network.

Next, we formalize the Stackelberg game as a bi-level optimization problem. For our problem, the upper level problem (ULP) corresponds to the FC who is the leader of the game, while the lower level problem (LLP) belongs to the attacker who is the follower.

$$\begin{aligned}
&\underset{\{\tilde{c}_k\}_{k=1}^{K}\in\mathbb{C}}{\text{maximize}} \quad D(\{\tilde{c}_k\}_{k=1}^{K}) \\
&\text{subject to} \quad \sum_{k=1}^{K}\tilde{c}_k N_k \leq C_{budget}^{network} \\
&\qquad\qquad \underset{B_k\in\mathbb{Z}^+}{\text{minimize}} \quad D(\{B_k\}_{k=1}^{K}) \\
&\qquad\qquad \text{subject to} \quad \sum_{k=1}^{K}\tilde{c}_k B_k \leq C_{budget}^{attacker} \\
&\qquad\qquad\qquad\qquad 0 \leq B_k \leq N_k, \forall\, k = 1, 2, \ldots, K
\end{aligned} \tag{5.17}$$

where $\mathbb{Z}^+$ is the set of non-negative integers. Notice that the bi-level optimization problem, in general, is an NP-hard problem. In fact, the LLP is a variant of the packing formulation of the bounded knapsack problem with a non-linear objective function. This is, in general, NP-hard. Using existing algorithms, cost set $\{\tilde{c}_k\}_{k=1}^{K}$ and attack configuration $\{B_k\}_{k=1}^{K}$ can be determined at the cost of computational efficiency. In this chapter, we identify a special case of the above problem which can be solved in polynomial time to determine the equilibrium strategies. To solve the bi-level optimization problem, we first solve the LLP assuming the solution of the ULP to be some fixed $(\tilde{c}_1, \cdots, \tilde{c}_K)$. This approach will give us a structure of the optimal $\{B_k\}_{k=1}^{K}$ for any arbitrary $\{\tilde{c}_k\}_{k=1}^{K}$. Next, using the structure of the optimal $\{B_k\}_{k=1}^{K}$, the bi-level optimization problem simplifies to finding the solution $\{\tilde{c_k}\}_{k=1}^{K}$ of the ULP. Finally, we present a polynomial time algorithm to solve the bi-level optimization problem, i.e., to find $\{\tilde{c_k}\}_{k=1}^{K}$ and, thus, $\{B_k\}_{k=1}^{K}$.

Next, we discuss the relationships that enable our problem to have a polynomial time solution.

We define profit $P(S)$ of an attack configuration $S = \{B_k\}_{k=1}^K$ as follows[9]

$$P(S) = D(\phi) - D(S) = D(\phi) - D(\{B_k\}_{k=1}^K),$$

where $D(\phi)$ is the KLD when there are no Byzantines in the network and $D(S) = D(\{B_k\}_{k=1}^K)$ is the KLD with $\{B_k\}_{k=1}^K$ Byzantines in the tree network. Next, we define the concept of dominance which will be used later to explore some useful properties of the optimal attack configuration $\{B_k\}_{k=1}^K$.

Definition 5.5.1. *We say that a set S_1 dominates another set S_2 if*

$$P(S_1) \geq P(S_2) \text{ and } C(S_1) \leq C(S_2), \tag{5.18}$$

where $P(S_i)$ and $C(S_i)$ denote the profit and cost incurred by using set S_i, respectively. If in (5.18)*, $P(S_1) > P(S_2)$, S_1 strictly dominates S_2 and if $P(S_1) = P(S_2)$, S_1 weakly dominates S_2.*

To solve the bi-level optimization problem, we first solve the LLP assuming the solution of the ULP to be some fixed $(\tilde{c}_1, \cdots, \tilde{c}_K)$. We refer to LLP as a maximum damage Byzantine attack problem. Observe that, knowing that the FC chooses $(\tilde{c}_1, \cdots, \tilde{c}_K)$, the LLP can be reformulated as follows:

$$\begin{aligned} \underset{B_k \in \mathbb{Z}^+}{\text{minimize}} \quad & \sum_{k=1}^{K} N_k D_k(\{B_i\}_{i=1}^k) \\ \text{subject to} \quad & \sum_{k=1}^{K} \tilde{c}_k B_k \leq C_{budget}^{attacker} \\ & 0 \leq B_k \leq N_k, \ \forall k = 1, \cdots, K. \end{aligned}$$

Next, we discuss the relationships that enable our maximum damage Byzantine attack problem to admit a polynomial time solution.

[9] In this section, we assume that the optimal operating point, i.e., (P_d^{k*}, P_{fa}^{k*}), is the same for all the nodes in the tree network. It has been shown that the use of identical thresholds is asymptotically optimal for parallel networks [103]. We conjecture that this result is valid for tree networks as well and employ identical thresholds.

5.5.1 Analysis of the Optimal Attack Configuration

In this section, we identify a special case of the bounded knapsack problem (LLP) which can be solved in polynomial time. More specifically, we show that if the set of defensive resources $\mathbb{C} = (c_1, \cdots, c_n)$ satisfy the cost structure $c_{max} \leq \left(\min_{k \in \{1,\cdots,K-1\}} \frac{N_{k+1}}{N_k} \right) \times c_{min}$[10] or $c_1 \leq \min_k a_k \times c_n$, then, the optimal solution $\{B_k\}_{k=1}^K$ exhibits the properties given in the lemma below.

Lemma 5.5.2. *Given a K level tree network with cost structure satisfying $c_{max} \leq \left(\min_{k \in \{1,\cdots,K-1\}} \frac{N_{k+1}}{N_k} \right) \times c_{min}$, the best response of an attacker with cost budget $C_{budget}^{attacker}$ is $\{B_k\}_{k=1}^K$ with*

$$B_1 = \left\lfloor \frac{C_{budget}^{attacker}}{\tilde{c}_1} \right\rfloor$$

and the remaining elements of B_k for $2 \leq k \leq K$ can be calculated recursively.

Proof. Please see Appendix A.9. □

It can also be shown that the solution $\{B_k\}_{k=1}^K$ will be non-overlapping and unique under the condition that the attacker cannot make $D_k = 0, \ \forall k$.

5.5.2 Bi-Level Optimization Algorithm

Based on Lemma 5.5.2, in this section we will present a polynomial time algorithm to solve the bi-level optimization problem, i.e., to find $\{\tilde{c_k}\}_{k=1}^K$ and $\{B_k\}_{k=1}^K$. Using the cost structure $c_{max} \leq \left(\min_k \frac{N_{k+1}}{N_k} \right) \times c_{min}$, the attack configuration $\{B_k\}_{k=1}^K$ as given in Lemma 5.5.2 can be determined in a computationally efficient manner. Due to the structure of the optimal $\{B_k\}_{k=1}^K$, the bi-level optimization problem simplifies to finding the solution $\{\tilde{c_k}\}_{k=1}^K$ of the ULP.

To solve this problem, we use an iterative elimination approach. We start by listing all $\binom{n}{K}$ combinations from the set $\mathbb{C}$, denoted as, $S = \{s_i\}_{i=1}^{\binom{n}{K}}$. Without loss of generality, we assume that the elements of $s_i = \{c_1^i, \cdots, c_K^i\}$ are arranged in descending order, i.e., $c_k^i \geq c_{k+1}^i, \forall k$. Notice

[10] Notice that, in the case of the perfect M-ary tree networks, the proposed cost structure simplifies to $c_{max} \leq M \times c_{min}$.

that, all these $\binom{n}{K}$ combinations will satisfy $c_k^i \leq \frac{N_{k+1}}{N_k} c_{k+1}^i$, because

$$c_k^i \leq c_{max} \leq \min_j \frac{N_{j+1}}{N_j} c_{min} \leq \min_j \frac{N_{j+1}}{N_j} c_{k+1}^i \leq \frac{N_{k+1}}{N_k} c_{k+1}^i.$$

Next, we discard all those subsets s_i from S which violate the network designer's cost budget constraint. If the set S is empty, then there does not exist any solution for the ULP. Otherwise, the problem reduces to finding the subset s_i which maximizes the KLD. To find the subset s_i which maximizes the KLD, using the dominance relationship we start with assigning the cost $\tilde{c_1} = \min_{k \in s} c_1^k$, where s has the elements which are solutions of $\arg\min_i \left\lfloor \frac{C_{budget}^{attacker}}{c_1^i} \right\rfloor$. Next, we discard all those subsets s_i from S which do not have $\tilde{c_1}$ as their first element and solve the problem recursively.

The pseudo code of the polynomial time algorithm to find $\{\tilde{c_k}\}_{k=1}^K$ and $\{B_k\}_{k=1}^K$ is presented as Algorithm 5.1.

5.5.3 An Illustrative Example

Let us consider a two-level network with $N_1 = 6$ and $N_2 = 12$. We assume that $\mathbb{C} = \{4,\ 3,\ 2\}$, $C_{budget}^{network} = 60$ and $C_{budget}^{attacker} = 11$. Next, we solve the bi-level optimization problem. Observe that, costs satisfy $c_1 \leq 2 \times c_3$. So the algorithm chooses the solution of the ULP as ($\tilde{c}_1 = 4$, $\tilde{c}_2 = 3$) and the solution of the LLP as ($B_1 = \left\lfloor \frac{11}{4} \right\rfloor = 2$, $B_2 = \left\lfloor \frac{11-2\times4}{3} \right\rfloor = 1$). To corroborate these results, in Figure 5.3, we plot the $\min_{P_{1,0}, P_{0,1}}$ KLD for all combinations of the parameters B_1 and B_2 in the tree. We vary the parameter B_1 from 0 to 6 and B_2 from 0 to 12. All the feasible solutions are plotted in red and unfeasible solutions are plotted in blue. Figure 5.3 corroborates the results of our algorithm.

Notice that, the attack configuration $\{B_k\}_{k=1}^K$ is the set containing the *number* of Byzantines residing at different levels of the tree. However, the FC cannot identify the Byzantines in the network. Also, notice that when the adversary attacks more than 50% of the nodes at level 1, the decision fusion scheme becomes completely incapable. In these scenarios, where the FC is blind, the knowledge of attack configuration will not incur any performance benefit. Next, we present a

Algorithm 5.1 Bi-Level Optimization Algorithm

Require: $\mathbb{C} = \{c_k\}_{k=1}^{n}$ with $c_{max} \leq \left(\min_j \frac{N_{j+1}}{N_j} \right) \times c_{min}$

$S \leftarrow$ All K out of n combinations $\{s_i\}_{i=1}^{\binom{n}{K}}$ with elements of s_i arranged in decreasing order

for $i = 1$ **to** $\binom{n}{K}$ **do**

  if $\sum_{k=1}^{K} c_k^i \times N_k > C_{budget}^{network}$ **then**

    $S \leftarrow S/s_i$

  end if

end for

if S is an empty set **then**

  return $(\phi,\ \phi)$

else

  for $k = 1$ **to** K **do**

    $\tilde{c}_k = \min_{j \in s} c_k^j$ where s has elements which are solutions of $\arg\min_i \left\lfloor \frac{C_{budget}^{attacker}}{c_k^i} \right\rfloor$

    $B_k \leftarrow \left\lfloor \frac{C_{budget}^{attacker}}{\tilde{c}_k} \right\rfloor$

    $C_{budget}^{attacker} \leftarrow (C_{budget}^{attacker} - \tilde{c}_k B_k)$

  end for

  return $(\{\tilde{c}_k\}_{k=1}^{K},\ \{B_k\}_{k=1}^{K})$

end if

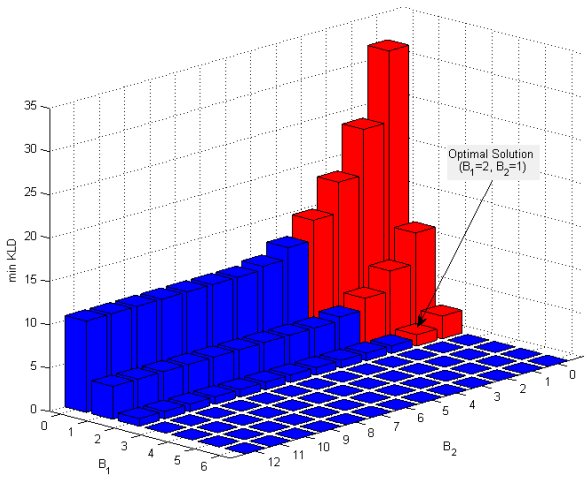


Fig. 5.3: min KLD vs. attack configuration (B_1, B_2) for $P_d = 0.9$, $P_{fa} = 0.1$.

reputation-based Byzantine identification/mitigation scheme, which works even when the network is blind, in order to improve the detection performance of the network. We propose a simple yet efficient Byzantine identification scheme and analyze its performance.

5.6 An Efficient Byzantine Identification Scheme

In this section, we propose and analyze a Byzantine identification scheme to be implemented at the FC.

5.6.1 Byzantine Identification Scheme

We assume that the FC has the knowledge of the attack model and utilizes this knowledge to identify the Byzantines. The FC observes the local decisions of each node over a time window T, which can be denoted by $(k, i) = [u_1(k, i), \ldots, u_T(k, i)]$ for $1 \leq i \leq N_k$ at level $1 \leq k \leq K$. We also assume that there is one honest anchor node with probability of detection P_d^A and probability of false alarm P_{fa}^A present and known to the FC. We employ the anchor node to provide the gold standard which is used to detect whether or not other nodes are Byzantines. The FC can also serve as an anchor node when it can directly observe the phenomenon and make a decision. We denote the Hamming distance between the reports of the anchor node and an honest node i at

level k over the time window T by $d_H^A(k,i) = ||U^A - U^H(k,i)||$, that is the number of elements that are different between U^A and $U^H(k,i)$. Similarly, the Hamming distance between reports of the anchor node and a Byzantine node i at level k over the time window T is denoted by $d_B^A(k,i) = ||U^A - U^B(k,i)||$. Since the FC is aware of the fact that Byzantines might be present in the network, it compares the Hamming distance of a node i at level k to a threshold η_k, $\forall i, \forall k$ (a procedure to calculate η_k is discussed later in the chapter), to make a decision to identify the Byzantines. In tree networks, a Byzantine node alters its decision as well as received decisions from its children prior to transmission in order to undermine the network performance. Therefore, solely based on the observed data of a node i at level k, the FC cannot determine whether the data has been flipped by the node i itself or by one of its Byzantine parent node. In our scheme, the FC makes the inference about a node being Byzantine by analyzing the data from the node i as well as its predecessor nodes' data. FC starts from the nodes at level 1 and computes the Hamming distance between reports of the anchor node and the nodes at level 1. FC declares node i at level 1 to be a Byzantine if and only if the Hamming distance of node i is greater than a fixed threshold η_1. Children of identified Byzantine nodes $\mathbb{C}(\mathbb{B}_1)$ are not tested further because of the non-overlapping condition. However, if a level 1 node is determined not to be a Byzantine, then, the FC tests its children nodes at level 2. The FC declares node i at level k, for $2 \leq k \leq K$, to be a Byzantine if and only if the Hamming distance of node i is greater than a fixed threshold η_k and Hamming distances of all predecessors of node i is less than equal to their respective thresholds η_j.

In this way, it is possible to counter the data falsification attack by isolating Byzantine nodes from the information fusion process. The probability that a Byzantine node i at level k is isolated at the end of the time window T, is denoted as $P_B^{iso}(k,i)$.

5.6.2 Performance Analysis

As mentioned earlier, local decisions of the nodes are compared to the decisions of the anchor node over a time window of length T. The probability that an *honest* node i at level k makes a decision

that is different from the anchor node is given by

$$\begin{aligned} & P^{AH}_{diff}(k,i) \\ = \;& P(u^A_i = 1, u^H_{k,i} = 0, H_0) + P(u^A_i = 0, u^H_{k,i} = 1, H_0) \\ & + P(u^A_i = 1, u^H_{k,i} = 0, H_1) + P(u^A_i = 0, u^H_{k,i} = 1, H_1) \\ = \;& P_0[(P^k_{fa} + P^A_{fa}) - 2P^k_{fa}P^A_{fa}] + P_1[(P^k_d + P^A_d) - 2P^k_d P^A_d] \\ \doteq \;& P_0[P^{AH}_{diff}(k,i,0)] + P_1[P^{AH}_{diff}(k,i,1)] \; . \end{aligned}$$

where the prior probabilities of the two hypotheses H_0 and H_1 are denoted by P_0 and P_1, respectively. The probability that a Byzantine node i at level k sends a decision different from that of the anchor node is given by

$$\begin{aligned} & P^{AB}_{diff}(k,i) \\ = \;& P(u^A_i = 1, u^B_{k,i} = 0, H_0) + P(u^A_i = 0, u^B_{k,i} = 1, H_0) \\ & + P(u^A_i = 1, u^B_{k,i} = 0, H_1) + P(u^A_i = 0, u^B_{k,i} = 1, H_1) \\ = \;& P_0[P^A_{fa}P^k_{fa} + (1 - P^A_{fa})(1 - P^k_{fa})] + P_1[P^A_d P^k_d + (1 - P^A_d)(1 - P^k_d)] \\ \doteq \;& P_0[P^{AB}_{diff}(k,i,0)] + P_1[P^{AB}_{diff}(k,i,1)] \; . \end{aligned}$$

The difference between the reports of a node and the anchor node under hypothesis $l \in \{0,1\}$ (i.e., $d^A_I(k,i,l),\ I \in \{H,B\}$) is a Bernoulli random variable with mean $P^{AH}_{diff}(k,i,l)$ for honest nodes and $P^{AB}_{diff}(k,i,l)$ for Byzantines. FC declares node i at level k to be a Byzantine if and only if the Hamming distance of node i is greater than a fixed threshold η_k and Hamming distances of all predecessors of node i are less than equal to their respective thresholds η_j. The probability that a Byzantine node i at level k is isolated at the end of the time window T can be expressed as

$$
\begin{aligned}
& P_B^{iso}(k,i) = P[(d_B^A(k,i) > \eta_k), (d_H^A(k-1,i) \leq \eta_{k-1}), \cdots, (d_H^A(1,i) \leq \eta_1)] \\
= & \sum_{l \in \{0,1\}} P_l \left[P[d_B^A(k,i,l) > \eta_k] \prod_{m=1}^{k-1} P[d_H^A(m,i,l) \leq \eta_m] \right] \\
= & \sum_{l \in \{0,1\}} P_l \left[\sum_{j=\eta_k+1}^{T} \binom{T}{j} (P_{diff}^{AB}(k,i,l))^j (1 - P_{diff}^{AB}(k,i,l))^{T-j} \prod_{m=1}^{k-1} \left[\sum_{j=0}^{\eta_m} \binom{T}{j} (P_{diff}^{AH}(m,i,l))^j (1 - P_{diff}^{AH}(m,i,l))^{T-j} \right] \right].
\end{aligned}
$$

For large T, by using the normal approximation, we get

$$
P_B^{iso}(k,i) = \sum_{l \in \{0,1\}} P_l \left[Q\left(\frac{\eta_k - TP_{diff}^{AB}(k,i,l)}{\sqrt{(TP_{diff}^{AB}(k,i,l)(1 - P_{diff}^{AB}(k,i,l)))}} \right) \prod_{m=1}^{k-1} Q\left(\frac{TP_{diff}^{AH}(m,i,l) - \eta_m}{\sqrt{(TP_{diff}^{AH}(m,i,l)(1 - P_{diff}^{AH}(m,i,l)))}} \right) \right]
$$

This can be written recursively as follows

$$
P_B^{iso}(k+1,i) = \sum_{l \in \{0,1\}} P_l \left[(1 - b(k,l)) \left(\frac{a(k+1,l)}{a(k,l)} \right) P_B^{iso}(k,i,l) \right], \tag{5.19}
$$

with $P_B^{iso}(k,i) \doteq \sum_{l \in \{0,1\}} P_l[P_B^{iso}(k,i,l)]$, and

$$
a(k,l) = Q\left(\frac{\eta_k - TP_{diff}^{AB}(k,i,l)}{\sqrt{(TP_{diff}^{AB}(k,i,l)(1 - P_{diff}^{AB}(k,i,l)))}} \right),
$$

$$
b(k,l) = Q\left(\frac{\eta_k - TP_{diff}^{AH}(k,i,l)}{\sqrt{(TP_{diff}^{AH}(k,i,l)(1 - P_{diff}^{AH}(k,i,l)))}} \right).
$$

One can choose η_k such that the isolation probability of honest nodes at level k based solely on its data under the hypothesis H_l (i.e., $b(k,l)$) is constrained to some value $\delta_k << 0.5$. In other words, we choose η_k such that $\max_{l \in \{0,1\}} b(k,l) = \delta_k$, i.e.,

$$
\eta_k = Q^{-1}(\delta_k) \sqrt{TP_{diff}^{AH}(k,i,l^*)(1 - P_{diff}^{AH}(k,i,l^*))} + TP_{diff}^{AH}(k,i,l^*) \tag{5.20}
$$

where $l^* = \arg\max_l b(k,l)$. Now, the expression for $a(k,l)$ can be written as

$$a(k,l) = Q\left(\frac{Q^{-1}(\delta_k)\sqrt{P^{AH}_{diff}(k,i,l^*)(1-P^{AH}_{diff}(k,i,l^*))} + \sqrt{T}(P^{AH}_{diff}(k,i,l^*) - P^{AB}_{diff}(k,i,l))}{\sqrt{P^{AB}_{diff}(k,i,l)(1-P^{AB}_{diff}(k,i,l))}}\right)$$

Now using the fact that $\max_l P^{AH}_{diff}(k,i,l) < \min_l P^{AB}_{diff}(k,i,l)$, it can be shown that $(P^{AH}_{diff}(k,i,l^*) - P^{AB}_{diff}(k,i,l)) < 0,\ \forall i$ and, therefore, $\lim_{T\to\infty} a(k,l) = 1$.

Lemma 5.6.1. *For a K level tree network, for our proposed Byzantine identification scheme, the asymptotic (i.e., $T \to \infty$) probability that a Byzantine node i at level $k+1$, for $1 \le k \le K-1$, is isolated is lower-bounded by,*

$$\prod_{j=2}^{k}(1-\delta_j).$$

Proof. Notice that, $\lim_{T\to\infty} a(k,l) = 1$. The asymptotic performance of the proposed scheme can be analyzed as follows:

$$\begin{aligned}
\lim_{T\to\infty} P^{iso}_B(k+1,i) &= \sum_{l\in\{0,1\}} P_l \lim_{T\to\infty}\left[(1-b(k,l))\left(\frac{a(k+1,l)}{a(k,l)}\right)P^{iso}_B(k,i,l)\right] \\
&\ge (1-\delta_k)\sum_{l\in\{0,1\}} P_l \lim_{T\to\infty}\left[P^{iso}_B(k,i,l)\right] \\
&= \prod_{j=2}^{k}(1-\delta_j).
\end{aligned}$$

□

Notice that, the parallel network topology is a special case of the tree network topology with $K = 1$. For $K = 1$, our scheme can identify all the Byzantines with probability one because $\lim_{T\to\infty} P^{iso}_B(1,i) = \lim_{T\to\infty}\sum_{l\in\{0,1\}} P_l[a(1,l)] = 1$. When $K > 1$, we can choose η_k appropriately such that Byzantines can be identified with a high probability.

Next, to gain insights into the solution, we present some numerical results in Figure 5.4 that corroborate our theoretical results. We consider a tree network with $K = 5$ and plot $P^{iso}_B(k,i)$, $1 \le$

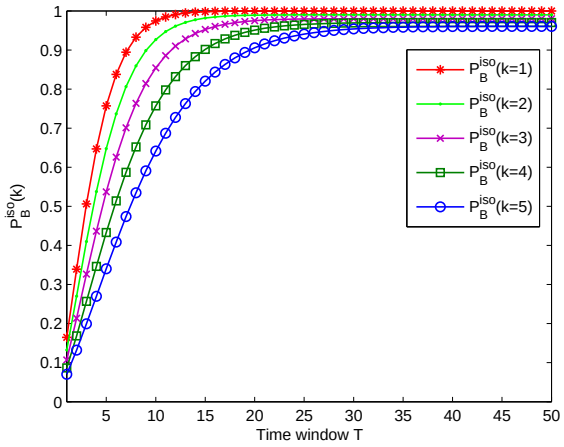


Fig. 5.4: Isolation probability $P_B^{iso}(k,i)$ vs. time window T.

$k \leq 5$, as a function of the time window T. We assume that the operating points (P_d^k, P_{fa}^k), $1 \leq k \leq 5$, for the nodes at different levels are given by $[(0.8, 0.1), (0.75, 0.1), (0.6, 0.1), (0.65, 0.1), (0.6, 0.1)]$ and for anchor node $(P_d^A, P_{fa}^A) = (0.9, 0.1)$. We also assume that the hypotheses are equi-probable, i.e., $P_0 = P_1 = 0.5$, and the maximum isolation probability of honest nodes at level k based solely on its data is constrained by $\delta_k = 0.01, \forall k$. It can be seen from Figure 5.4 that in a span of only $T = 25$ time windows, our proposed scheme isolates/identifies almost all the Byzantines in the tree network.

5.7 Discussion

In this chapter, we considered the problem of optimal Byzantine attacks on distributed detection mechanism in tree networks. We analyzed the performance limit of detection performance with Byzantines and obtained the optimal attacking strategies that minimize the detection error exponent. The problem was also studied from the network designer's perspective. It was shown that the optimal local detector is independent of the Byzantine's parameter. Next, we modeled the strategic interaction between the FC and the attacker as a Leader-Follower (Stackelberg) game and attacker and defender (FC) equilibrium strategies were identified. We also proposed a simple yet efficient scheme to identify Byzantines and analytically evaluated its performance.

CHAPTER 6
FAULT TOLERANT DISTRIBUTED INFERENCE: TREE TOPOLOGY

6.1 Introduction

Detection, classification, or estimation of certain events, targets, or phenomena, in a region of interest, is an important application of inference networks. As discussed in Chapter 1, there have been limited attempts to address distributed inference problems in tree networks [30, 47, 48, 99, 128]. In all but the simplest cases, optimal strategies in tree based networks are difficult to derive. Most of the work on tree networks focuses on person-by-person optimal (PBPO) strategies [47, 48, 99, 128]. Also, the above works address the problem of distributed detection in tree networks while, to the best of our knowledge, the problem of distributed estimation in tree networks has not received any attention. Due to the complexity of classification and estimation in tree networks as compared to detection, these problems have been left unexplored by researchers.

In this chapter, we take a first step to address the distributed inference (classification and estimation) problems in tree networks by developing an analytically tractable framework, proposing efficient algorithms and carrying out asymptotic analysis to characterize performance. We propose the use of coding-theory based techniques to solve the problem of fault tolerant distributed

inference (both classification and estimation) in tree networks. Next, we propose computationally efficient algorithms for designing the optimal code matrices used in these schemes. We also study the asymptotic inference performance of our schemes for two different classes of tree networks: fixed height tree networks and fixed degree tree networks, and prove the asymptotic optimality of the proposed schemes under certain conditions. Finally, we also show the robustness of the coding-theory proposed schemes using simulations.

The remainder of the chapter is organized as follows. In Section 6.2, we describe the system architecture and present a brief overview of Distributed Classification Fusion using Error Correcting Codes (DCFECC) scheme [117] which serves as a foundation for the schemes presented in this chapter. We propose our basic coding scheme for distributed classification in tree networks in Section 6.3. The performance of the proposed scheme in the asymptotic regime is also analyzed. We present some numerical results to gain insights into the solution. We extend this scheme for distributed estimation in tree networks in Section 6.4 by formulating the estimation problem as a sequence of M-ary classification problems. The performance of the proposed scheme in the asymptotic regime is analyzed and some numerical results are presented. We also provide a technique for optimal region splitting for distributed estimation. Finally, we conclude our chapter in Section 6.5 with some discussion on possible future work.

6.2 Preliminaries

6.2.1 General Network Architecture

Consider a perfect tree, $T(K,\ N)$, rooted at the FC. Nodes at level k, for $1 \leq k \leq K-1$, are referred to as intermediate nodes and nodes at the last level of the tree, i.e., $k = K$, are called the leaf nodes. In a perfect tree, all the intermediate nodes have an equal number of immediate successors and all leaf nodes are at the same depth. The number of such immediate successors N is referred to as the degree of the tree.

We assume that the network is designed to infer about a particular phenomenon. Each node j

at level k performs two basic operations:

- Depending on the task, senses data regarding the phenomenon and/or collects data from its successors at level $k+1$, denoted by $S^{k+1}(j)$.
- Compresses the data available at node j about the phenomenon and transmits a 1-bit version to its predecessor at level $k-1$, denoted by $P^{k-1}(j)$.

Local observation of node j at level k is denoted as y_j^k. Received data vector at node j of level k from its successors $S^{k+1}(j)$ at level $k+1$ is denoted as $\mathbf{v_j^k} \in \{0,1\}^N$. After processing the data at the node according to a processing model (Please see Figure 6.1), every node j at level k sends its one-bit local decision $u_j^k \in \{0,1\}$ to its immediate predecessor. This processing model is designed based on the inference problem considered, i.e., Figure 6.2 for classification or Figure 6.5 for estimation. Finally, the FC receives the inference vector $\mathbf{u^1} = (u_1^1, \cdots, u_N^1) \in \{0,1\}^N$ and fuses this data to infer about the underlying phenomenon. In our analysis, we consider error-free links in the network. However, we do provide some simulation results for the case where there are erroneous links, to examine the robustness of the proposed schemes.

Given a tree network, our objective is to find the appropriate processing scheme for nodes at all levels depending on the inference problem considered. DCFECC scheme as explaind in Sec. 2.4 (originally proposed for parallel topology in [117]) which serves as the mathematical basis for the ideas proposed in this chapter.

6.3 Distributed Classification in Tree Networks

In this section, we consider the problem of distributed classification in tree networks. We model the classification problem as an M-ary hypotheses testing problem. Let H_l, where $l = 1, \cdots, M$ and $M \geq 2$, denote the M hypotheses[1]. The *a priori* probabilities of these M hypotheses are denoted by $Pr(H_l) = P_l$, for $l = 1, \cdots, M$.

[1] In order to distinguish among the M hypotheses using binary decisions, we assume that $N \geq \log_2 M$.

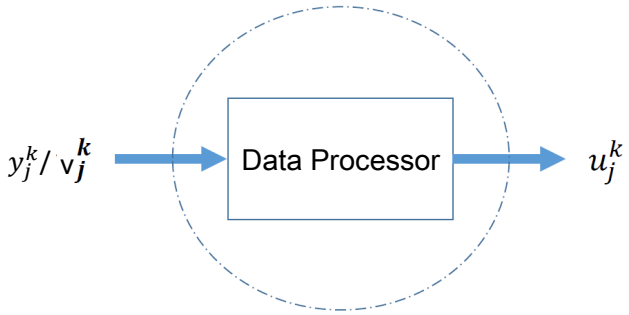


Fig. 6.1: Data processing for distributed inference at node j at level k. Here y_j^k and $\mathbf{v_j^k}$ are the inputs and $u_j^k \in \{0,1\}$ is the output of the process at node j.

6.3.1 Proposed Scheme

We assume that under each hypothesis H_l, every leaf node j acts as a source and makes an independent and identically distributed (i.i.d.) observation y_j^K. After processing the observations locally, every leaf node j sends its local decision[2] $u_j^K \in \{0,1\}$ according to a transmission mapping $\tau_j^K(\cdot)$ to its immediate predecessor $P^{K-1}(j)$. Each intermediate node j at level k receives the decision vector consisting of local decisions made by its immediate successors $S^{k+1}(j)$ at level $k+1$, which can be expressed as $\mathbf{v_j^k} = \mathbf{u^{k+1}} = (u_1^{k+1}, \cdots, u_N^{k+1})$. After fusing this data using fusion rule $f_j^k(\cdot)$, this intermediate node j at level k makes a classification decision $y_j^k \in \{1, \cdots, M\}$. Then, it sends a 1-bit version of this decision, $u_j^k \in \{0,1\}$, according to its transmission mapping $\tau_j^k(\cdot)$ to its immediate predecessor $P^{k-1}(j)$. Finally, the FC receives the decision vector $\mathbf{u^1} = (u_1^1, \cdots, u_N^1)$ and fuses this data to decide the underlying hypothesis. The proposed scheme builds on the DCFECC scheme (Section 2.4). To summarize, each node j at level k, for $1 \leq k \leq K-1$, performs two basic operations (Please see Figure 6.2):

- Collects data from its successors $S^{k+1}(j)$ and fuses their data using fusion rule $f_j^k(\cdot)$ to locally decide the hypothesis, denoted by y_j^k.
- Compresses the decision y_j^k at node j about the hypothesis and transmits a 1-bit version u_j^k to its predecessor $P^{k-1}(j)$ using the transmission mapping $\tau_j^k(\cdot)$.

[2]In this context, “decision" is a binary quantized value determined by the processing model.

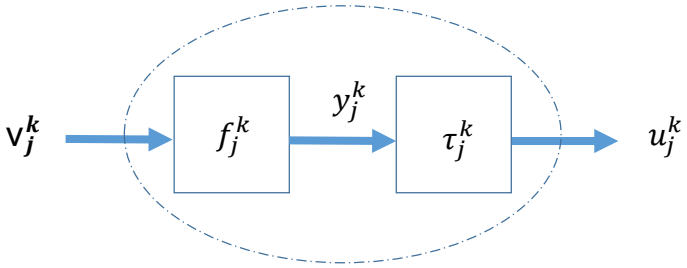


Fig. 6.2: Data processing for distributed classification at node j at level $1 \leq k \leq K-1$. Here $\mathbf{v_j^k} \in \{0,1\}^N$, $y_j^k \in \{1, \cdots, M\}$, and $u_j^k \in \{0,1\}$. Therefore, the mappings are $f_j^k : \{0,1\}^N \rightarrow \{1, \cdots, M\}$ and $\tau_j^k : \{1, \cdots, M\} \rightarrow \{0,1\}$

For the leaf nodes (level K), there are no successors and, therefore, only the second operation needs to be performed. And, for the FC (level '0'), only the first operation needs to be performed. Each of the functions $f_j^k(\cdot)$ and $\tau_j^k(\cdot)$ depend on the code matrix used at level k. By appealing to symmetry, we assume that each node at the same level k, uses an identical code matrix C^k for transmission to its predecessor and C^{k+1} for fusion of data from its successors. We start with the design of transmission mapping $\tau_j^K(\cdot)$ of the leaf nodes. When the leaf nodes use code matrix C^K for transmission, the probability of misclassification at level $K-1$ is given by [117]

$$P_e^{K-1} = \sum_{\mathbf{i},l} \int_{\mathbf{y^K}} P_l P(u_1^K = i_1 | y_1^K) \times \cdots \times P(u_N^K = i_N | y_N^K) p(\mathbf{y^K} | H_l) \psi_{\mathbf{i},l}^K, \tag{6.1}$$

where $\mathbf{i} = [i_1, \cdots, i_N] \in \{0,1\}^N$ is a realization of the received codeword $\mathbf{u^K}$, $\mathbf{y^K} = [y_1^K, \cdots, y_N^K]$ are the local observations of leaf nodes, and $\psi_{\mathbf{i},l}^K$ is the cost associated with a global decision H_l at level $K-1$ when the received vector from level K is $\mathbf{i}$. This cost is:

$$\psi_{\mathbf{i},l}^k = \begin{cases} 1 - \frac{1}{\varrho} & \text{if } \mathbf{i} \text{ is in the decision region of } H_l \\ 1 & \text{otherwise.} \end{cases} \tag{6.2}$$

for $k = K$, where ϱ is the number of decision regions corresponding to a received codeword $\mathbf{i}$. In other words, it is the number of rows of code matrix C^K which have the same minimum Hamming distance with the received codeword $\mathbf{i}$. Usually this value is 1, however ϱ can be greater than one

when there is a tie at the node at level $K-1$ and in those cases, the tie-breaking rule is to choose one of them randomly.

Employing a person-by-person optimization approach, we can find the local transmission mapping of the leaf nodes as follows [117]:

$$u_j^K = \tau_j^K(y_j^K) = \begin{cases} 0, & \text{if } \sum_l p(y_j^K|H_l)A_{jl} < 0 \\ 1, & \text{otherwise} \end{cases}, \tag{6.3}$$

where $A = \{A_{jl}\}$ is the weight matrix whose values[3] are given by,

$$A_{jl} = \sum_{i_1,\cdots,i_{j-1},i_{j+1},\cdots,i_N} P_l P(u_1^K = i_1|H_l) \times \cdots \times P(u_{j-1}^K = i_{j-1}|H_l) P(u_{j+1}^K = i_{j+1}|H_l) \\ \times \cdots \times P(u_N^K = i_N|H_l) \times [\psi^K_{i_1,\cdots,i_{j-1},0,i_{j+1},\cdots,i_N,l} - \psi^K_{i_1,\cdots,i_{j-1},1,i_{j+1},\cdots,i_N,l}]. \tag{6.4}$$

For $1 \leq k \leq K-1$, the local classification decision $y_j^k \in \{1,\cdots,M\}$ made using the data from the successors $S^{k+1}(j)$ is discrete and, therefore, the transmission mapping $\tau^k(\cdot)$ is straightforward and is given as follows:

$$u_j^k = \tau_j^k(y_j^k) = c^k_{y_j^k j}, \quad \text{if } 1 \leq k \leq K-1. \tag{6.5}$$

In other words, the one-bit decision u_j^k is the element of C^k corresponding to y_j^kth row and jth column. For every intermediate node, the fusion rule $f_j^k(\cdot)$ is the minimum Hamming distance fusion rule as given in (2.8). Therefore, the performance of the scheme depends on the minimum Hamming distance of the code matrices. Let d_{min}^k be the minimum Hamming distance of the code matrix C^k. Pseudo code of the proposed scheme for distributed classification is presented as Algorithm 6.1.

[3] We refer the reader to [117] for further details.

Algorithm 6.1 Distributed Classification in Tree Networks

Require: $N \geq \log_2 M$, code matrices C^k for $k = 1, \cdots, K$

1: Every leaf node j acquires an observation y_j^K

2: Process y_j^K using (6.3) and send $u_j^K \in \{0, 1\}$ according to code matrix C^K

3: **for** each intermediate node j at level $k = K - 1$ to 1 **do**

4: Collect $\mathbf{u^{k+1}} = (u_1^{k+1}, \cdots, u_N^{k+1})$

5: Fuse $\mathbf{u^{k+1}}$ using C^{k+1} to make a classification decision $y_j^k \in \{1, \cdots, M\}$

6: Send 1-bit version of the classification decision $u_j^k \in \{0, 1\}$ according to code matrix C^k to intermediate predecessor

7: **end for**

8: FC collects and fuses $\mathbf{u^1}$ using code matrix C^1 to decide underlying hypothesis

In the remainder of this section, we derive the error expressions at intermediate nodes which will later be used for the design of code matrices at every level.

Proposition 6.3.1. *The probability of misclassification P_e^{k-1} at level $k-1$ due to the data received from level k and using code matrix $C^k = \{c_{mj}^k\}$ ($1 \leq k \leq K-1$, $1 \leq m \leq M$, $1 \leq j \leq N$) is:*

$$P_e^{k-1} = \sum_{\mathbf{i},l} P_l \prod_{j=1}^{N} \left[(2i_j - 1) \sum_{m=1}^{M} c_{mj}^k P_{ml}^k + (1 - i_j) \right] \psi_{\mathbf{i},l}^k, \tag{6.6}$$

where $\mathbf{i} = [i_1, \cdots, i_N] \in \{0, 1\}^N$ is the realization of the received codeword $\mathbf{u^k}$, matrix $P^k = \{P_{ml}^k\}$ is the confusion matrix of the local decisions at level k, and $\psi_{\mathbf{i},l}^k$ is the cost associated with a global decision H_l at level $k-1$ when the received vector from level k is **i**. *This cost is given by* (6.2).

Proof. If u_j^k denotes the bit sent by the node j at level k and the global decision is made using the Hamming distance criterion:

$$P_e^{k-1} = \sum_{\mathbf{i},l} P_l P(\mathbf{u^k} = \mathbf{i} | H_l) \psi_{\mathbf{i},l}^k. \tag{6.7}$$

Since local decisions are conditionally independent, $P(\mathbf{u^k} = \mathbf{i}|H_l) = \prod_{j=1}^{N} P(u_j^k = i_j|H_l)$. Further,

$$\begin{aligned} P(u_j^k = i_j|H_l) &= i_j P(u_j^k = 1|H_l) + (1 - i_j)P(u_j^k = 0|H_l) \\ &= (1 - i_j) + (2i_j - 1)P(u_j^k = 1|H_l) \\ &= (1 - i_j) + (2i_j - 1)\sum_{m=1}^{M} c_{mj}^k P(y_j^k = m|H_l) \\ &= (1 - i_j) + (2i_j - 1)\sum_{m=1}^{M} c_{mj}^k P_{ml}^k \end{aligned}$$

where y_j^k is the local classification decision made by node j after collecting data from its successors $S^{k+1}(j)$ at level $k + 1$. The desired result follows. □

Note that this suggests that the probability of misclassification at level $k - 1$ is dependent on the confusion matrix at level k. These can be derived easily as follows:

$$P_{ml}^k \triangleq P(\text{decide } H_m \text{ at level } k|H_l \text{ is true}) = 1 - \sum_{\mathbf{i}} p(\mathbf{u^k} = \mathbf{i}|H_l)\psi_{\mathbf{i},m}^{k+1} \tag{6.8}$$

From these expressions, we can observe that there is a recursive structure, where the probability of misclassification at level k is dependent on the confusion matrix of level $k + 1$. Therefore, the performance at the FC depends on all the code matrices in a recursive manner. As mentioned before, we propose a simpler approach by assuming that each node of the same level uses the same code matrix which is designed by optimizing on a person-by-person sequential basis. We start with the code design at level $K - 1$ to fuse data from level K. This is designed by optimizing the expression in (6.1). Once we have designed the optimal code matrix at this level, we derive the corresponding confusion matrix from (6.8), which is used to design the code matrix at the next level by optimizing the expression in (6.6). Following this method, we can design all the code matrices. Note that each of these optimizations can be performed offline using approaches such as

simulated annealing or cyclic-column replacement [117]. In the following subsection, we analyze our scheme in the asymptotic regime and show that the scheme is asymptotically optimal.

6.3.2 Asymptotic Optimality

We study the asymptotic classification performance of our scheme for two different classes of tree networks. The first one is the class of *fixed height trees* in which the height of the tree, K, is assumed to be fixed while the second is the class of *fixed degree trees* in which the degree of the tree, N, is assumed to be fixed. More specifically, we study the classification performance of minimum Hamming distance fusion in fixed height tree networks when the number of nodes tends to infinity and in fixed degree tree networks when the height of the tree tends to infinity. We first provide the following bound on the misclassification probability at the FC which will be used to prove the asymptotic optimality. Let Q_m^k be the probability of misclassifying hypothesis H_m at level k and define $q_{max}^k \stackrel{\Delta}{=} \max_{1\leq m\leq M} Q_m^k$. Note that for levels $0 \leq k \leq K-1$, we have $Q_m^k = 1 - P_{mm}^k$ where P_{ml}^k are the elements of the confusion matrix. For $k = K$, $Q_m^K = 1 - Pr(\text{decide } H_m \text{ at level } K | H_m \text{ is true})$.

Proposition 6.3.2. *In a perfect tree structure $T(K,N)$ employing the proposed scheme, the misclassification probability at the FC, P_e^0, is bounded as follows*

$$P_e^0 \leq \left[q_{max}^K\right]^{\prod_{k=1}^{K} \frac{d_{min}^k}{a_k}}, \tag{6.9}$$

if $q_{max}^K < \frac{1}{2}$ and

$$d_{min}^k \geq \frac{2(M-2)}{[1-4q_{max}^k(1-q_{max}^k)] - (1/a_k)[(2/q_{max}^k)-2]}, \ \forall k, \tag{6.10}$$

where a_k is a parameter which satisfies the following condition

$$a_k > \frac{2(1-q_{max}^k)}{q_{max}^k - 4(q_{max}^k)^2(1-q_{max}^k)}, \forall k. \tag{6.11}$$

Proof. Pleas see Appendix A.10. □

The results obtained in Proposition 6.3.2 show that the misclassification probability for minimum Hamming distance fusion can be upper-bounded by a quantity determined by the minimum Hamming distance of the code matrices ($d_{min}^k,\ \forall k$), and the largest local classification error among all hypotheses ($q_{max}^k,\ \forall k$). Also, note that the parameter a_k in (6.9) can be chosen appropriately to make the bound tighter. For example, if a_k is chosen such that

$$\frac{q_{max}^k(2M-2)+2(1-q_{max}^k)}{q_{max}^k-4(q_{max}^k)^2(1-q_{max}^k)} > a_k > \frac{2(1-q_{max}^k)}{q_{max}^k-4(q_{max}^k)^2(1-q_{max}^k)}, \forall k,$$

then, $(d_k/a_k) > 1, \forall k$, and we have

$$P_e^0 \leq \left[q_{max}^K\right]^{\prod_{k=1}^K (d_{min}^k/a_k)} \leq \left[q_{max}^K\right]^{(d_{min}^K/a_K)}.$$

As a consequence, for fixed height trees, the decoding error of the proposed scheme vanishes as d_{min}^K approaches infinity which happens when $N \to \infty$. Also, for fixed degree trees, the decoding error of the proposed scheme vanishes as K approaches infinity. These results can be summarized in the following theorem:

Theorem 6.3.3. *Under conditions* (6.10) *and* (6.11)*, the proposed coding theory based distributed classification scheme is asymptotically optimal, for both classes of tree networks: fixed height tree networks and fixed degree tree networks, as long as the probabilities of correct local classification for all hypotheses of the leaf nodes are greater than one half.*

The conditions required for the above theorem depend on the minimum Hamming distance d_{min}^k of the code matrices used at each level and can be interpreted as follows: the proposed scheme is optimal when the minimum Hamming distance of the code matrices is “large enough" to ensure that perfect classification is made at every level of the tree. When the rows of the code matrices are well separated due to large minimum Hamming distance, the proposed scheme can handle more errors and have good performance.

Also, observe that these results imply that when (6.10) and (6.11) are satisfied, there is no loss in asymptotic performance when all the nodes at level k, for $k = 1, \cdots, K$, use identical transmission mapping and identical fusion rules.

6.3.3 Simulation Results

In this section, we evaluate the performance of the proposed scheme using simulations. Consider a tree network $T(3, 7)$ consisting of a total $N_{total} = 400$ nodes, including the FC. The leaf nodes sense the environment to identify among four ($M = 4$) equally likely hypotheses. As discussed before, we assume that all the leaf node measurements are independent and identically distributed. Under each hypothesis, the probability density function is assumed to be Gaussian distribution with the same variance ($\sigma^2 = 1$) but with different means $0, s, 2s$, and $3s$ respectively. The signal-to-noise power ratio (SNR) of observations at each local node is given by $20 \log_{10} s$. The code matrices are designed using the scheme described in Section 6.3.1 and simulated annealing for optimization. The designed code matrices used at different levels of the tree are found to be

$$C^1 = [11, 8, 9, 9, 3, 9, 12] \tag{6.12}$$

$$C^2 = [7, 6, 3, 12, 12, 9, 14] \tag{6.13}$$

$$C^3 = [3, 8, 14, 12, 9, 12, 9] \tag{6.14}$$

where the code matrix is represented by a vector of M-bit integers. Each integer m_j represents a column of any arbitrary code matrix C and can be expressed as $m_j = \sum_{l=1}^{M} c_{lj}$. For example, the integer 9 in column 5 of C^3 represents $c^3_{15} = 1, c^3_{25} = 0, c^3_{35} = 0$, and $c^3_{45} = 1$.

In Figure 6.3, we plot the final probability of misclassification at the FC with varying SNR values. Note that this probability of misclassification is empirically found by performing $N_{mc} = 5000$ Monte-Carlo runs. As we can observe, the performance of the scheme improves with increasing SNR and approaches 0 as early as 5dB.

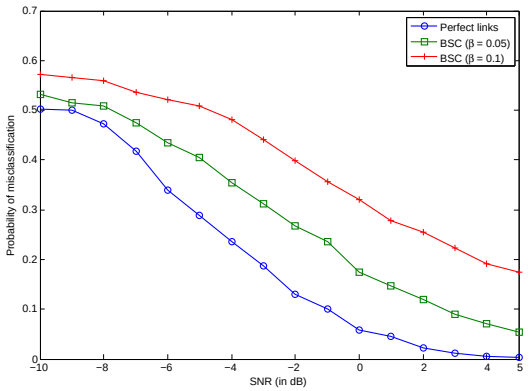


Fig. 6.3: Probability of misclassification versus SNR

Also, in order to show the benefits of fusion, we have provided the misclassification probability values at the intermediate nodes in Table 6.1. It can be seen from Table 6.1 that the misclassification probability decreases with each level showing the benefits of fusion at every level.

Table 6.1: Misclassification probability at intermediate nodes for a 3-level tree

SNR (in dB)	Level 2	Level 1	Level 0 (FC)
-10	0.585	0.5137	0.493
-5	0.5057	0.4062	0.3072
0	0.3371	0.1991	0.0662
5	0.138	0.03826	0.0022

Since the proposed scheme is based on error-correcting codes, it can also tolerate some errors in data. These errors could be due to various reasons: presence of a faulty node [117], presence of imperfect links between levels [16], or presence of a malicious node sending falsified data [108]. In order to check the fault-tolerance capability of the scheme, we have simulated the case when the links between the levels are binary symmetric channels with crossover probabilities $\beta = 0.05$ and $\beta = 0.1$[4]. As shown in Figure 6.3, the proposed scheme still performs reasonably well even in the presence of imperfect data due to non-ideal channels modeled as binary symmetric channels.

Building on these results, in the following section, we address the parameter estimation problem in tree based networks. More specifically, we break the parameter estimation problem into a sequence of M-ary decision making problems, and each of these M-ary decision making problems

[4]The code design here is based on the channel-unaware approach where the code is designed for ideal channels.

is solved using a technique similar to the distributed classification scheme of the previous section.

6.4 Distributed Parameter Estimation in Tree Networks using Iterative Classification

Consider a distributed parameter estimation problem where the goal is to estimate a random scalar parameter θ at the FC. The parameter θ has a prior probability density function (pdf) $p_\theta(\theta)$ where $\theta \in \Theta$. We propose a scheme to estimate the parameter θ using iterative classification. By doing so, we break the parameter estimation problem into a sequence of M-ary decision making problems. This is essentially a process of iterative rejection of unlikely objects where the most undesirable options are discarded and the scope of options is progressively narrowed down until exactly one option is left.

6.4.1 Proposed Scheme

We consider a distributed estimation system with the topology of a perfect tree, $T(K,\ N)$, rooted at the FC. We model the parameter estimation problem as an M-ary hypotheses testing problem. Our scheme is iterative in which at every iteration $1 \leq s \leq K$, the parameter space is split into M regions and an M-ary hypothesis test is performed at the level $(K+1-s)$ of the tree to determine the parameter space for the next level in the tree. The optimal splitting of the parameter space at every iteration can be determined offline (which will be explained later in the chapter in Section 6.4.3). For now, we assume that the M^K final regions and their corresponding representation points are known. Let H_l^k, where $l = 1, \cdots, M$ and $M \geq 2$, denote the M hypotheses[5] being tested at level k. Figure 6.4 shows an example of parameter space splitting when $p_\theta(\cdot)$ is standard Gaussian, and $M = K = 2$. Every node at level $k = 2$ first performs a classification task to determine if the parameter θ is positive or negative (differentiate between hypotheses H_1^2 and H_2^2). After a decision is made, the nodes at level $k = 1$, 'zoom' into the decided hypothesis, say H_1^2,

[5] As before, we assume that $N \geq \log_2 M$.

and perform a classification task to determine if θ belongs to hypothesis H_1^2 or H_2^1. In this manner, the FC at level '0' eventually decides the true hypothesis, among the M^K hypotheses, where θ belongs.

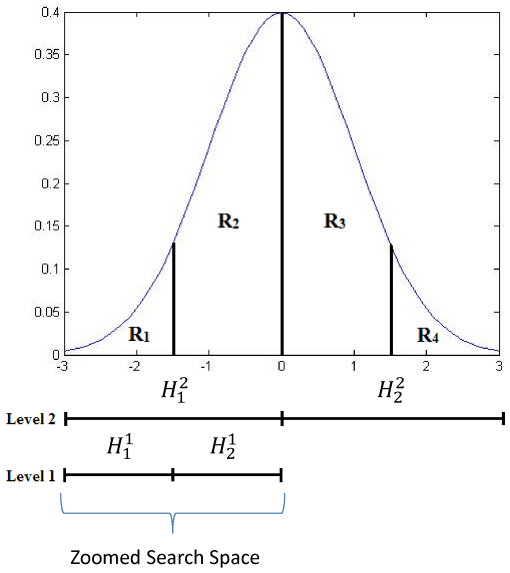


Fig. 6.4: An example of splitting of parameter space.

The *a priori* probabilities of the M^K hypotheses are denoted by $Pr(H_l^k) = P_l^k$, for $l = 1, \cdots, M$ and $k = 1, \cdots, K$. P_l^k depends on $p_\theta(\cdot)$ and the region corresponding to H_l^k. We assume that every node j' at level $k+1$ acts as a source and makes a conditionally independent and identically distributed (i.i.d.) observation $y_{j'}^{k+1}$, conditioned under each hypothesis H_l^k. After processing the observations locally, every node sends its local decision $u_{j'}^{k+1} \in \{0, 1\}$ according to a transmission mapping $\tau^{k+1}(\cdot)$ to its immediate predecessor $P^k(j')$. Each intermediate node j at level k receives the decision vector $\mathbf{v_j^k}$ consisting of local decisions made by its immediate successors $S^{k+1}(j)$ at level $k+1$. Intermediate nodes at level k, through collaboration[6] and fusion, decide on the result of the M-ary hypotheses test as the new parameter space for them.

The scheme builds on the DCFECC scheme proposed for distributed classification. Each node

[6]In collaboration phase, node j at level k shares $\mathbf{v_j^k}$ (the data collected from its successors $S^{k+1}(j)$) with other nodes at level k. In this chapter, we assume that nodes do not compress $\mathbf{v_j^k}$ for collaboration and, therefore, after collaboration phase receive the data $\mathbf{v^k} = \left[\mathbf{v_1^k}, \cdots, \mathbf{v_N^k}\right] \in \{0, 1\}^{N^k}$.

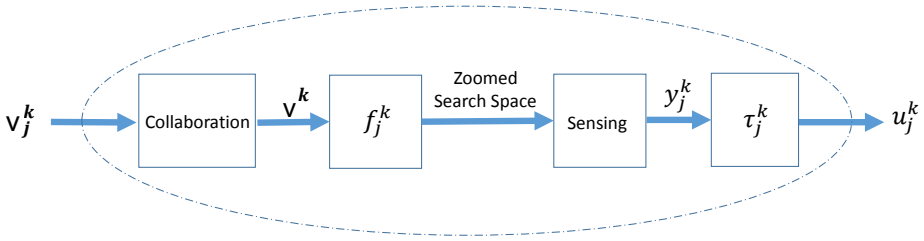


Fig. 6.5: Data processing for distributed estimation at node j at level $1 \leq k \leq K-1$. Here, $\mathbf{v_j^k} \in \{0,1\}^N$, $\mathbf{v^k} \in \{0,1\}^{N^k}$, $y_j^k \in \mathbb{R}$, and $u_j^k \in \{0,1\}$. Therefore, the mappings are $f_j^k : \{0,1\}^{N^k} \rightarrow \{1, \cdots, M\}$ and $\tau_j^k : \mathbb{R} \rightarrow \{0,1\}$.

j at level k, for $1 \leq k \leq K-1$ performs four basic operations (Please see Figure 6.5):

- Collects data from its successors $S^{k+1}(j)$ and collaborates with other nodes at level k.
- Decides the new parameter space (hypotheses to test) by fusing data using fusion rule $f_j^k(\cdot)$.
- Acquires observation y_j^k and performs hypothesis testing to determine the new parameter space.
- Compresses the observation at node j about the hypothesis (new parameter space) and transmits a 1-bit version to the predecessor $P^{k-1}(j)$ using the transmission mapping $\tau_j^k(\cdot)$.

For the leaf nodes (level K), there are no successors and, therefore, only the third and fourth operations need to be performed. The FC (level '0') collects data from its successors and makes the final decision regarding the region where θ belongs. Given a tree network, our objective is to find efficient transmission mappings and fusion rules for nodes at all levels, to maximize the estimation performance at the FC.

Remark: There are three major differences between the scheme proposed here for distributed estimation in tree networks and the scheme proposed in Section 6.3 for distributed classification in tree networks:

- In the scheme proposed here, every node acts as a source node and senses the phenomenon while for the classification problem in Section 6.3, only the leaf nodes act as source nodes and intermediate nodes act only as relay nodes.

- In Section 6.3, each node performs the same classification task, or in other words, the set of classes are the same. On the other hand, in the scheme proposed here, the parameter space is 'zoomed' at every level, which changes the corresponding classes to be tested.

- An important step in the scheme proposed in this section is the collaboration step which is not required for the classification problem of Section 6.3.

By appealing to symmetry, we assume that each node at the same level k, uses an identical code matrix C^k for transmission to its predecessor and C^{k+1} for fusion of data from its successors. Each of the functions $f_j^k(\cdot)$ and $\tau_j^k(\cdot)$ depend on the code matrix used at level k. Although the performance metric in this framework is the Mean Square Error (MSE), it is difficult to obtain a closed form representation for MSE. Therefore, typically, one uses the bounds on MSE to characterize the performance of the estimator. Here, we use an analytically tractable metric to analyze the performance of the proposed scheme which is the probability of misclassification of the parameter region. It is an important metric when the final goal of the parameter estimation task is to find the approximate region or neighborhood where the parameter lies rather than the true value of the parameter itself. Since the final region could be one of the M^K regions, a metric of interest is the probability of 'zooming' into the correct region. In other words, it is the probability that the true value of the parameter and the estimated value of the parameter lie in the same region.

Now, we design the transmission mapping $\tau_j^k(\cdot)$ of nodes at level k. Notice that the final region of the estimated value of the parameter is the same as the true value of the parameter, if and only if we 'zoom' into the correct region at every iteration of the proposed scheme. Thus, when the nodes at level k use code matrix C^k for transmission, the probability of misclassification at level $k-1$ is given by the following proposition.

Proposition 6.4.1. *The probability of misclassification P_e^{k-1} at level $k-1$ due to the data received from level k and using code matrix $C^k = \{c_{mj}^k\}$ ($1 \leq k \leq K$, $1 \leq m \leq M$, $1 \leq j \leq N^k$) is*

$$P_e^{k-1} = 1 - \prod_{t=k}^{K} \left[1 - \sum_{\mathbf{i},l} \int_{\mathbf{y^t}} P_l^t P(u_1^t = i_1 | y_1^t) \times \cdots \times P(u_{N^t}^t = i_{N^t} | y_{N^t}^t) p(\mathbf{y^t} | H_l^t) \psi_{\mathbf{i},l}^t \right], \quad (6.15)$$

where $\mathbf{i} = [i_1, \cdots, i_{N^t}] \in \{0,1\}^{N^t}$ *is the realization of the received codeword* $\mathbf{u^t}$, $\mathbf{y^t} = [y_1^t, \cdots, y_{N^t}^t]$ *are the local observations of nodes at level* t, *matrix* $P^t = \{P_{ml}^t\}$ *is the confusion matrix of the local decisions at level* t, *and* $\psi_{\mathbf{i},l}^{t-1}$ *is the cost associated with a global decision* H_l^{t-1} *at level* $t-1$ *when the received vector from level* t *is* $\mathbf{i}$. *This cost is:*

$$\psi_{\mathbf{i},l}^t = \begin{cases} 1 - \frac{1}{\varrho} & \text{if } \mathbf{i} \text{ is in decision region of } H_l^t \\ 1 & \text{otherwise.} \end{cases} \quad (6.16)$$

where as before ϱ *is the number of decision regions corresponding to a received codeword* $\mathbf{i}$. *In other words, it is the number of rows of code matrix* C^t *which have the same minimum Hamming distance with the received codeword* $\mathbf{i}$. P_l^t *is the prior probability of hypothesis* H_l^t *at level* t.

Proof. Note that a correct decision is made at level $k-1$ if and only if the decision at all levels from $t = k$ to $t = K$ are correct. Therefore, using (6.1) in a recursive manner at every level of the tree, we get the desired result. □

From (6.15), we can observe that the performance at the FC depends on all the code matrices in a recursive manner. In this chapter, we employ a simpler approach by assuming that code matrices are designed by optimizing on a person-by-person basis. The code matrix at each level is designed using an approach similar to that of a parallel topology. Note that each of these optimizations can be performed offline using approaches such as simulated annealing or cyclic-column replacement [117].

Employing a person-by-person optimization approach, we can find the local transmission map-

ping of the nodes at level k as follows:

$$u_j^k = \tau_j^k(y_j^k) = \begin{cases} 0, & \text{if } \sum_l p(y_j^k|H_l^k)A_{jl}^k < 0 \\ 1, & \text{otherwise} \end{cases}, \tag{6.17}$$

where $A^k = \{A_{jl}^k\}$ is a weight matrix whose values are given by,

$$A_{jl}^k = \sum_{i_1,\cdots,i_{j-1},i_{j+1},\cdots,i_{N^k}} P_l^k P(u_1^k = i_1|H_l^k) \times \cdots \times P(u_{j-1}^k = i_{j-1}|H_l^k) P(u_{j+1}^k = i_{j+1}|H_l^k) \\ \times \cdots \times P(u_{N^k}^K = i_{N^k}|H_l^k) \times [\psi_{i_1,\cdots,i_{j-1},0,i_{j+1},\cdots,i_{N^k},l}^k - \psi_{i_1,\cdots,i_{j-1},1,i_{j+1},\cdots,i_{N^k},l}^k]. \tag{6.18}$$

For every intermediate node, the fusion rule $f_j^k(\cdot)$ is the minimum Hamming distance fusion rule as given in (2.8). Therefore, the performance of the scheme depends on the minimum Hamming distance of the code matrices. Let d_{min}^k be the minimum Hamming distance of the code matrix C^k.

Pseudo code of the proposed scheme for distributed estimation is presented as Algorithm 6.2.

In the remainder of this section, we analyze our scheme in the asymptotic regime and show that the scheme is asymptotically optimal.

6.4.2 Asymptotic Optimality

As before, we study the asymptotic performance of our scheme for two different classes of tree networks, *fixed height trees* and *fixed degree trees*. We also analyze the scenarios where both the number of nodes and the height of the tree tend to infinity. We first provide the following bound on the misclassification probability at the FC which will be used to prove the asymptotic optimality. Let Q_m^k be the probability of misclassifying hypothesis H_m^k at level k and define $q_{max}^k \stackrel{\Delta}{=} \max_{1\le m\le M} Q_m^k$. For $k = 1, \cdots, K$, $Q_m^K = 1 - Pr(\text{decide } H_m^k \text{ at level } K|H_m^k \text{ is true})$.

Algorithm 6.2 Distributed Estimation in Tree Networks

Require: $N \geq \log_2 M$, code matrices C^k for $k = 1, \cdots, K$

1: Every leaf node j acquires an observation y_j^K and performs hypothesis test to determine new parameter space
2: Process y_j^K using (6.17) and send $u_j^K \in \{0, 1\}$ according to code matrix C^K
3: **for** each intermediate node j at level $k = K - 1$ to 1 **do**
4: Collect $\mathbf{v_j^k}$ and collaborate with other nodes at level k to acquire $\mathbf{v^k} = \left[\mathbf{v_1^k}, \cdots, \mathbf{v_N^k}\right]$
5: Decide new parameter space by fusing $\mathbf{v^k}$ using C^{k+1}
6: Acquire y_j^k and perform hypothesis test to determine new parameter space
7: Process y_j^k using (6.17) and send $u_j^k \in \{0, 1\}$ according to code matrix C^k
8: **end for**
9: FC collects and fuses $\mathbf{u^1}$ using code matrix C^1 to decide the region where θ belongs
10: Estimate $\hat{\theta}$ is given by the representation point of the final decision region

Proposition 6.4.2. *In a perfect tree structure $T(K, N)$ employing the proposed scheme, if $q_{max}^k < \frac{1}{2}$, the misclassification probability at the FC, P_e^0, is bounded as follows*

$$P_e^0 \leq 1 - \prod_{k=1}^{K} \left[1 - (M-1)(4q_{max}^k(1 - q_{max}^k))^{\frac{d_{min}^k}{2}}\right]. \tag{6.19}$$

Proof. Please See Appendix A.11. □

As a consequence of Proposition (6.4.2), for fixed height trees, the probability of 'zooming' into the incorrect region of the proposed scheme vanishes as d_{min}^k approaches infinity which happens when $N \to \infty$.

$$
\begin{aligned}
& \lim_{N\to\infty} P_e^0 \\
\leq\ & \lim_{N\to\infty} \left[1 - \prod_{k=1}^{K} \left(1 - (M-1)(4q_{max}^k(1-q_{max}^k))^{\frac{d_{min}^k}{2}}\right)\right] \\
=\ & \left[1 - \prod_{k=1}^{K} \lim_{N\to\infty} \left(1 - (M-1)(4q_{max}^k(1-q_{max}^k))^{\frac{d_{min}^k}{2}}\right)\right] \\
=\ & 1 - \prod_{k=1}^{K} \left[1 - (M-1) \lim_{N\to\infty} \left((4q_{max}^k(1-q_{max}^k))^{\frac{d_{min}^k}{2}}\right)\right] \\
=\ & 1 - \prod_{k=1}^{K} [1 - (M-1)0] \\
=\ & 0.
\end{aligned}
$$

Hence, the overall detection probability becomes '1' as the degree of the tree N goes to infinity. This shows that the proposed scheme asymptotically attains perfect region detection probability for bounded height tree networks if $q_{max}^k < 1/2\ \forall k = 1, \cdots, K$. Notice that perfect region detection probability does not imply that the estimation error will vanish. It just provides a coarse estimate of the parameter. For estimation error to vanish, $M^K \to \infty$, which can be achieved by letting K approach infinity.

However, for fixed degree trees, misclassification error of the proposed scheme need not vanish as K approaches infinity.

$$
\begin{aligned}
& \lim_{K\to\infty} P_e^0 \\
\leq\ & \lim_{K\to\infty} \left[1 - \prod_{k=1}^{K} \left(1 - (M-1)(4q_{max}^k(1-q_{max}^k))^{\frac{d_{min}^k}{2}}\right)\right] \\
=\ & \left[1 - \lim_{K\to\infty} \prod_{k=1}^{K} \left(1 - (M-1)(4q_{max}^k(1-q_{max}^k))^{\frac{d_{min}^k}{2}}\right)\right]
\end{aligned}
$$

For misclassification error to vanish, every term in the product should vanish, which obviously

is not true for the above equation.

These results can be summarized as the following theorem:

Theorem 6.4.3. *The proposed iterative classification scheme for distributed parameter estimation in tree based networks is asymptotically optimal (when both the degree N and number of levels K simultaneously approach infinity), as long as the probabilities of correct local classification for all hypotheses at each node is greater than one half.*

Remark: Note that, while for distributed classification, we have shown that the proposed scheme is asymptotically optimal if either N or K tend to infinity, for the distributed estimation case, we have proved that the scheme is asymptotically optimal when both N and K tend to infinity.

Next, we address the remaining aspect of the scheme which is the discretization of the continuous parameter space to perform estimation as iterative classification.

6.4.3 Optimal Splitting of the Parameter Space

As mentioned before, the scheme splits the parameter space Θ into M^K regions. Therefore, the MSE between the true parameter value θ and the FC's estimate $\hat{\theta}$ is affected by two factors: the quantization of the continuous region Θ into M^K discrete points and the probability of misclassifying the region where the true parameter belongs. In Section 6.4.2, we showed that the probability of misclassification can be made to tend to zero by using a large sensor network. Therefore, in order to minimize the MSE, we need to minimize the error due to the quantization of Θ into M^K points. This optimal splitting depends on the prior pdf $p_\theta(\cdot)$ and can be determined by using ideas from rate distortion theory [23]. As mentioned in [23], the optimal regions for quantization are given by Voronoi regions and the reconstruction points should minimize the conditional expected distortion over their respective assignments. One of the most popular algorithms used to determine these regions is the Llyod-Max algorithm [63, 67]. This algorithm is iterative where we start with an initial set of reconstruction points which are typically chosen at random. It then repeatedly executes the following steps until convergence:

- Compute the optimal set of reconstruction regions (Voronoi regions) and
- Find the set of optimal reconstruction points for these regions (centroid of the Voronoi regions).

In this chapter, we use this algorithm which is performed offline and, therefore, is not a computational issue.

6.4.4 Simulation Results

In this section, we provide simulation results to evaluate the performance of the proposed scheme. As before, consider a tree network $T(3,7)$ consisting of a total $N_{total} = 400$ nodes, including the FC. The observation at each node is Gaussian distributed with unknown mean θ and variance σ^2. This unknown parameter θ is uniformly distributed in $(0, \theta_{max})$ where the region size is varied by varying the maximum value θ_{max}. At each level, the nodes perform an M-ary classification where $M = 4$. Therefore, there are a total of $M^K = 4^3 = 64$ possible estimates of θ. Since the parameter is uniformly distributed, the optimal splitting is uniform quantization into M^K regions with the mid-points of the regions as the corresponding representation. Due to the complexity in designing the optimal matrix of size 4×343 for transmission at level 3 (due to collaboration, each node at level 2 has data of all nodes at level 3), we employ a sub-optimal approach by concatenating the optimal code matrix of size 4×7. For level 3, it is concatenated 49 times, and for transmission at level 2, it is concatenated 7 times. The smaller code matrix of size 4×7 is designed using the simulated annealing approach.

In Figure 6.6, we plot the mean square error (MSE) between the true value of θ and its estimate $\hat{\theta}$ at the FC[7] as a function of θ_{max} when $\sigma^2 = 1$. This value of MSE is empirically found by performing $N_{mc} = 5000$ Monte-Carlo runs. As we can observe, the performance of the scheme gets worse with increasing region size. This is because, when the range of θ is increased while the total number of possible estimates remains fixed, the error due to quantization increases. Since

[7] As discussed before, this estimate is one of the M^K discrete points representing the quantized regions (centroids of the Voronoi regions, please see Section 6.4.3).

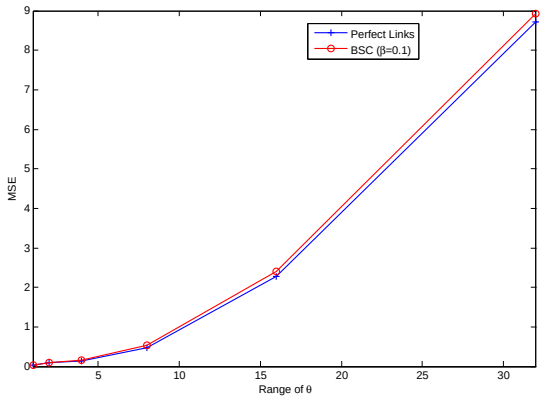


Fig. 6.6: MSE as a function of the range of θ

the proposed scheme is based on error-correcting codes, it can tolerate some errors in data. As mentioned before, these errors could be due to various reasons. We have also simulated the case when the links between the levels are modeled as binary symmetric channels with crossover probability $\beta = 0.1$. As shown in Figure 6.6, the proposed scheme is quite robust to the presence of imperfect data arising due to non-ideal channels modeled as binary symmetric channels. As alluded to before, this robustness in performance is due to the use of error-correcting codes. Similar observations can be made from Figure 6.7 where we plot MSE of the proposed estimation scheme as a function of observation variance σ^2 when $\theta_{max} = 32$.

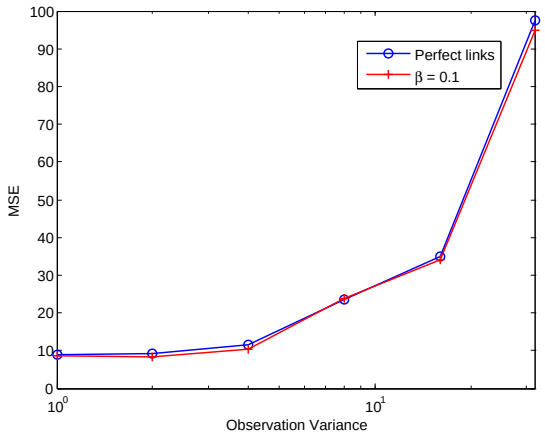


Fig. 6.7: MSE of the proposed estimation scheme with varying observation variance

6.5 Discussion

In this chapter, we considered the general framework of distributed inference in tree networks. We proposed an analytically tractable scheme to solve these problems and proved the asymptotic optimality of the proposed schemes. For the classification problem, when the number of hypotheses is $M = 2$, the proposed scheme is a majority-vote scheme for distributed detection in tree networks. Also, note that since the proposed scheme uses error-correcting codes, it works well even in scenarios with unreliable data. It should be pointed out that the proposed scheme is not limited to wireless sensor networks, although the application of wireless sensor networks has been considered in this chapter. The DCFECC scheme has been found to be applicable to a number of other applications including the paradigm of crowdsourcing. We believe that one can use these results to address several other applications involving tree structures.

CHAPTER 7
DISTRIBUTED DETECTION WITH CORRUPTED DATA: PEER TO PEER TOPOLOGY

7.1 Introduction

In the previous chapters, the problem of distributed inference in corrupted data in the parallel and tree topology was studied. It was assumed that there exists a centralized FC to fuse the data and to make a global decision. However, in many scenarios, a centralized FC may not be available or the FC may become an information bottleneck causing degradation of system performance, potentially leading to system failure. Also, due to the distributed nature of future communication networks, and various practical constraints, e.g., absence of the FC, transmit power or hardware constraints and dynamic nature of the wireless medium, it may be desirable to employ alternate peer-to-peer local information exchange in order to reach a global decision. One such decentralized approach for peer-to-peer local information exchange and inference is the use of a consensus algorithm. This chapter considers the problem of signal detection in distributed networks in the presence of data falsification (Byzantine) attacks. Detection approaches considered in the chapter

are based on fully distributed consensus algorithms, where all of the nodes exchange information only with their neighbors in the absence of a fusion center. For such networks, we first characterize the negative effect of Byzantines on the steady-state and transient detection performance of conventional consensus-based detection algorithms. To avoid performance deterioration, we propose a distributed weighted average consensus algorithm that is robust to Byzantine attacks. We show that, under reasonable assumptions, the global test statistic for detection can be computed locally at each node using our proposed consensus algorithm. We exploit the statistical distribution of the nodes' data to devise techniques for mitigating the influence of data falsifying Byzantines on the distributed detection system. Since some parameters of the statistical distribution of the nodes' data might not be known a priori, we propose learning based techniques to enable an adaptive design of the local fusion or update rules.

The rest of the chapter is organized as follows. In Sections 7.2 and 7.3, we introduce our system model and Byzantine attack model, respectively. In Section 7.4, we study the security performance of weighted average consensus-based detection schemes. In Section 7.5, we propose a protection mechanism to mitigate the effect of data falsification attacks on consensus-based detection schemes. Finally, Section 7.6 concludes the chapter.

7.2 System model

Consider two hypotheses H_0 (signal is absent) and H_1 (signal is present). Also, consider N nodes organized in an undirected graph G which faces the task of determining which of the two hypotheses is true. We model the network topology as an undirected graph $G = (V, E)$, where $V = \{v_1, \cdots, v_N\}$ represents the set of nodes in the network with $|V| = N$. The set of communication links in the network correspond to the set of edges E, where $(v_i, v_j) \in E$, if and only if there is a communication link between v_i and v_j so that, v_i and v_j can directly communicate with

each other. The adjacency matrix A of the graph is defined as

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

The neighborhood of a node i is defined as

$$\mathcal{N}_i = \{v_j \in V : (v_i, v_j) \in E\}, \forall i \in \{1, 2, \cdots, N\}.$$

The degree d_i of a node v_i is the number of edges in E which include v_i as an endpoint, i.e., $d_i = \sum_{j=1}^{N} a_{ij}$.

The degree matrix D is defined as a diagonal matrix with $\text{diag}(d_1, \cdots, d_N)$ and the Laplacian matrix L is defined as

$$l_{ij} = \begin{cases} d_i & \text{if } j = i, \\ -a_{ij} & \text{otherwise.} \end{cases}$$

In other words, $L = D - A$. As an illustration, consider a network with six nodes trying to reach consensus (see Figure 2.1(c)). The degree matrix for this network is given by $D = \text{diag}(1, 3, 2, 4, 1, 1)$. The adjacency matrix A for this network is given by

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

Therefore, the Laplacian matrix $L = D - A$ for this network is given by

$$L = \begin{bmatrix} 1 & -1 & 0 & 0 & 0 & 0 \\ -1 & 3 & -1 & -1 & 0 & 0 \\ 0 & -1 & 2 & -1 & 0 & 0 \\ 0 & -1 & -1 & 4 & -1 & -1 \\ 0 & 0 & 0 & -1 & 1 & 0 \\ 0 & 0 & 0 & -1 & 0 & 1 \end{bmatrix}.$$

The consensus-based distributed detection scheme usually contains three phases: sensing, information fusion, and decision making. In the sensing phase, each node acquires the summary statistic about the phenomenon of interest. In this chapter, we adopt the energy detection method so that the local summary statistic is the received signal energy. Next, in the information fusion phase, each node communicates with its neighbors to update their state values (summary statistic) and continues with the consensus iteration until the whole network converges to a steady state which is the global test statistic. Finally, in the decision making phase, nodes make their own decisions about the presence of the phenomenon using this global test statistic. In the following each of these phases is described in more detail.

7.2.1 Sensing Phase

We consider an N-node network using the energy detection scheme [28]. For the ith node, the sensed signal z_i^t at time instant t is given by

$$z_i^t = \begin{cases} n_i^t, & \text{under } H_0 \\ \zeta_i s^t + n_i^t & \text{under } H_1, \end{cases}$$

where ζ_i is the deterministic gain corresponding to the sensing channel, s^t is the deterministic signal at time instant t, n_i^t is AWGN, i.e., $n_i^t \sim \mathcal{N}(0, \sigma_i^2)$ (where $\mathcal{N}$ denotes the normal distribution) and independent across time. Each node i calculates a summary statistic Y_i over a detection interval of M samples, as

$$Y_i = \sum_{t=1}^{M} |z_i^t|^2$$

where M is determined by the time-bandwidth product [28]. Since Y_i is the sum of the squares of M i.i.d. Gaussian random variables, it can be shown that $\frac{Y_i}{\sigma_i^2}$ follows a central chi-square distribution with M degrees of freedom (χ_M^2) under H_0, and, a non-central chi-square distribution with M degrees of freedom and parameter η_i under H_1, i.e.,

$$\frac{Y_i}{\sigma_i^2} \sim \begin{cases} \chi_M^2, & \text{under } H_0 \\ \chi_M^2(\eta_i) & \text{under } H_1 \end{cases}$$

where $\eta_i = E_s|\zeta_i|^2/\sigma_i^2$ is the local SNR at the ith node and $E_s = \sum_{t=1}^{M} |s^t|^2$ represents the sensed signal energy over M detection instants. Note that the local SNR is M times the average SNR at the output of the energy detector, which is $\frac{E_s|\zeta_i|^2}{M\sigma_i^2}$.

7.2.2 Information Fusion Phase

In this section, we give a brief introduction to conventional consensus algorithms [75]. and explain how consensus is reached using the following two steps.

Step 1: All nodes establish communication links with their neighbors, and broadcast their information state, $x_i(0) = Y_i$.

Step 2: Each node updates its local state information by a local fusion rule (weighted combination of its own value and those received from its neighbors) [75]. We denote node i's updated information at iteration k by $x_i(k)$. Node i continues to broadcast information $x_i(k)$ and update its local information state until consensus is reached. This process of updating information state can be written in a compact form as

$$x_i(k+1) = x_i(k) + \frac{\epsilon}{w_i} \sum_{j \in \mathcal{N}_i} (x_j(k) - x_i(k)) \tag{7.1}$$

where ϵ is the time step and w_i is the weight given to node i's information. Using the notation $x(k) = [x_1(k), \cdots, x_N(k)]^T$, network dynamics can be represented in the matrix form as,

$$x(k+1) = Wx(k)$$

where, $W = I - \epsilon$ diag$(1/w_1, \cdots, 1/w_N)L$ is referred to as a Perron matrix. The consensus algorithm is nothing but a local fusion or update rule that fuses the nodes' local information state with information coming from neighbor nodes, and it is well known that every node asymptotically reaches the same information state for arbitrary initial values [75].

7.2.3 Decision Making Phase

The final information state x^* after reaching consensus for the above consensus algorithm will be the weighted average of the initial states of all the nodes [75] or $x^* = \sum_{i=1}^{N} w_i Y_i / \sum_{i=1}^{N} w_i$, $\forall i$. Average consensus can be seen as a special case of weighted average consensus with $w_i = w, \ \forall i$. After the whole network reaches a consensus, each node makes its own decision about the hypothesis using a predefined threshold λ[1]

$$\text{Decision} = \begin{cases} H_1 & \text{if } x^* > \lambda \\ H_0 & \text{otherwise} \end{cases}$$

where weights are given by [127]

$$w_i = \frac{\eta_i/\sigma_i^2}{\sum_{i=1}^{N} \eta_i/\sigma_i^2}. \tag{7.2}$$

In the rest of the chapter, $\Lambda = \sum_{i=1}^{N} w_i Y_i / \sum_{i=1}^{N} w_i$ is referred to as the final test statistic.

Next, we discuss Byzantine attacks on consensus-based detection schemes and analyze the performance degradation of the weighted average consensus-based detection algorithms due to these attacks.

[1] In practice, parameters such as threshold λ and consensus time step ϵ are set off-line based on well know techniques [75]. In this chapter, these parameters are assumed known and setting the parameters is not considered.

7.3 Attacks on Consensus based Detection Algorithms

When there are no adversaries in the network, we noted in the last section that consensus can be reached to the weighted average of arbitrary initial values by having the nodes use the update strategy $x(k+1) = Wx(k)$ with an appropriate weight matrix W. However, suppose, that instead of broadcasting the true summary statistic Y_i and applying the update strategy (7.1), some nodes (referred to as Byzantines) deviate from the prescribed strategies. Accordingly, Byzantines can attack in two ways: data falsification (nodes falsify their initial data or weight values) and consensus disruption (nodes do not follow the update rule given by (7.1)). More specifically, Byzantine node i can do the following

$$\begin{aligned} &\text{Data falsification:} && x_i(0) = Y_i + \Delta_i, \quad \text{or} \quad w_i \text{ is changed to } \tilde{w}_i \\ &\text{Consensus disruption:} && x_i(k+1) = x_i(k) + \frac{\epsilon}{w_i} \sum_{j \in \mathcal{N}_i} (x_j(k) - x_i(k)) + u_i(k), \end{aligned}$$

where $(\Delta_i, \tilde{w}_i)$ and $u_i(k)$ are introduced at the initialization step and at the update step k, respectively. The attack model considered above is extremely general, and allows Byzantine node i to update its value in a completely arbitrary manner (via appropriate choices of $(\Delta_i, \tilde{w}_i)$, and $u_i(k)$, at each time step). An adversary performing consensus disruption attack has the objective to disrupt the consensus operation. However, consensus disruption attacks can be easily detected because of the nature of the attack. Furthermore, the identification of consensus disruption attackers has been investigated in the past literature (e.g., see [78, 94]) where control theoretic techniques were developed to identify disruption attackers in a 'single' consensus iteration. However, these techniques cannot identify the data falsification attacker due to philosophically different nature of the problem. Also, notice that, by knowing the existence of such an identification mechanism, a smart adversary will aim to disguise itself while degrading the detection performance. In contrast to disruption attackers, data falsification attackers are more capable and can manage to disguise themselves while degrading the detection performance of the network by falsifying their data. Susceptibility and protection of consensus strategies to data falsification attacks has received scant attention, and this

is the focus of our work. Our main focus[2] here is on the scenarios where an attacker performs a data falsification attack by introducing $(\Delta_i, \tilde{w}_i)$ during initialization. We exploit the statistical distribution of the initial values and devise techniques to mitigate the influence of Byzantines on the distributed detection system. Our approach for data falsification attacks on consensus-based detection systems complements the techniques proposed in [78, 94] that are mainly focused on consensus disruption attacks.

7.3.1 Data Falsification Attack

In data falsification attacks, attackers try to manipulate the final test statistic (i.e., $\Lambda = \sum_{i=1}^{N} w_i Y_i / \sum_{i=1}^{N} w_i$) in a manner so as to degrade the detection performance. We consider a network with N nodes that uses Algorithm (7.1) for reaching consensus. Weight w_i, given to node i's data Y_i in the final test statistic, is controlled or updated by node i itself while carrying out the iteration in (7.1). So by falsifying initial values Y_i or weights w_i, the attackers can manipulate the final test statistic. Detection performance will be degraded because Byzantine nodes can always set a higher weight to their manipulated information. Thus, the final statistic's value across the whole network will be dominated by the Byzantine node's local statistic that will lead to degraded detection performance.

Next, we define a mathematical model for data falsification attackers. We analyze the degradation in detection performance of the network when Byzantines falsify their initial values Y_i for fixed arbitrary weights $\tilde{w}_i$.

7.3.2 Attack Model

The objective of Byzantines is to degrade the detection performance of the network by falsifying their data (Y_i, w_i). We assume that Byzantines have an advantage and know the true hypothesis. Under this assumption, we analyze the detection performance of the data fusion schemes which yields the maximum performance degradation that the Byzantines can cause, i. e., worst case de-

[2]Later, we also come up with a robust distributed weighted average consensus algorithm which allows the detection of consensus disruption attack while mitigating the effect of data falsification attacks.

tection performance. We consider the case when weights of the Byzantines have been tampered by setting their value at $\tilde{w}_i$ and analyze the effect of falsifying the initial values Y_i. Now a mathematical model for a Byzantine attack is presented. Byzantines tamper with their initial values Y_i and send $\tilde{Y}_i$ such that the detection performance is degraded.

Under H_0:

$$\tilde{Y}_i = \begin{cases} Y_i + \Delta_i & \text{with probability } P_i \\ Y_i & \text{with probability } (1 - P_i) \end{cases}$$

Under H_1:

$$\tilde{Y}_i = \begin{cases} Y_i - \Delta_i & \text{with probability } P_i \\ Y_i & \text{with probability } (1 - P_i) \end{cases}$$

where P_i is the attack probability and Δ_i is a constant value which represents the attack strength, which is zero for honest nodes. As we show later, Byzantine nodes will use a large value of Δ_i so that the final statistic's value is dominated by the Byzantine node's local statistic leading to a degraded detection performance. We use deflection coefficient [54] to characterize the security performance of the detection scheme due to its simplicity and its strong relationship with the global detection performance. Deflection coefficient of the global test statistic is defined as: $\mathcal{D}(\Lambda) = \frac{(\mu_1 - \mu_0)^2}{\sigma^2_{(0)}}$, where $\mu_k = \mathbb{E}[\Lambda|H_k],\ k = 0, 1,$ is the conditional mean and $\sigma^2_{(k)} = \mathbb{E}[(\Lambda - \mu_k)^2|H_k],\ k = 0, 1,$ is the conditional variance. The deflection coefficient is closely related to performance measures such as the Receiver Operating Characteristics (ROC) curve [54]. In general, the detection performance monotonically increases with an increasing value of the deflection coefficient. We define the critical point of the distributed detection network as the minimum fraction of Byzantine nodes needed to make the deflection coefficient of the global test statistic equal to zero (in which case, we say that the network becomes *blind*) and denote it by α_{blind}. We assume that the communication between nodes is error-free and our network topology is fixed during the whole consensus process and, therefore, consensus can be reached without disruption [75].

In the next section, we analyze the security performance of consensus-based detection schemes

$$\mu_0 = \sum_{i=1}^{N_1} \left[P_i \frac{\tilde{w}_i}{\sum w} (M\sigma_i^2 + \Delta_i) + (1 - P_i) \frac{\tilde{w}_i}{\sum w} M\sigma_i^2 \right] + \sum_{i=N_1+1}^{N} \left[\frac{w_i}{\sum w} M\sigma_i^2 \right] \tag{7.3}$$

$$\mu_1 = \sum_{i=1}^{N_1} \left[P_i \frac{\tilde{w}_i}{\sum w} ((M + \eta_i)\sigma_i^2 - \Delta_i) + (1 - P_i) \frac{\tilde{w}_i}{\sum w} (M + \eta_i)\sigma_i^2 \right] + \sum_{i=N_1+1}^{N} \left[\frac{w_i}{\sum w} (M + \eta_i)\sigma_i^2 \right] \tag{7.4}$$

$$\sigma_{(0)}^2 = \sum_{i=1}^{N_1} \left(\frac{\tilde{w}_i}{\sum w} \right)^2 \left[P_i(1 - P_i)\Delta_i^2 + 2M\sigma_i^4 \right] + \sum_{i=N_1+1}^{N} \left(\frac{w_i}{\sum w} \right)^2 2M\sigma_i^4 \tag{7.5}$$

---

in the presence of data falsifying Byzantines as modeled above. This analysis will be useful in revealing some quantitative relationships to judge the degradation of the detection performance with data falsifying Byzantines.

7.4 Performance Analysis of Consensus-based Detection Algorithms

In this section, we analyze the effect of data falsification attacks on conventional consensus-based detection algorithms.

First, we characterize the effect of Byzantines on the steady-state performance of the consensus-based detection algorithms and determine α_{blind}.

Without loss of generality, we assume that the nodes corresponding to the first N_1 indices $i = 1, \cdots, N_1$ are Byzantines and the remaining nodes corresponding to indices $i = N_1+1, \cdots, N$ are honest nodes. Let us define $w = [\tilde{w}_1, \cdots, \tilde{w}_{N_1}, w_{N_1+1} \cdots, w_N]^T$ and $\sum w = \sum_{i=1}^{N_1} \tilde{w}_i + \sum_{i=N_1+1}^{N} w_i$.

Lemma 7.4.1. *For data fusion schemes, the condition to blind the network or equivalently to make the deflection coefficient zero is given by*

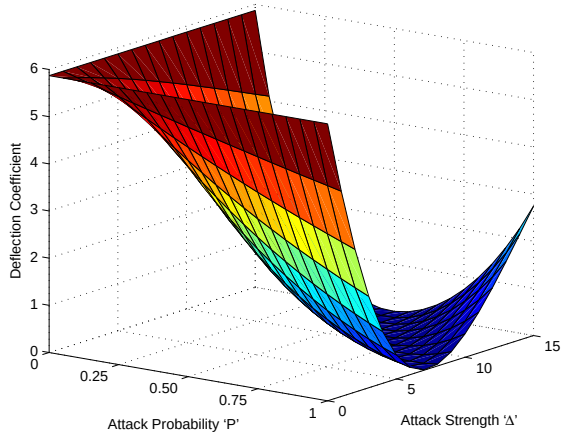


Fig. 7.1: Deflection Coefficient as a function of attack parameters P and Δ.

$$\sum_{i=1}^{N_1} \tilde{w}_i(2P_i\Delta_i - \eta_i\sigma_i^2) = \sum_{i=N_1+1}^{N} w_i\eta_i\sigma_i^2.$$

Proof. Please see Appendix A.12. □

Note that, when $w_i = \tilde{w}_i = z, \eta_i = \eta, \sigma_i = \sigma, P_i = P, \Delta_i = \Delta, \ \forall i$, the blinding condition simplifies to $\dfrac{N_1}{N} = \dfrac{1}{2}\dfrac{\eta\sigma^2}{P\Delta}$.

Next, to gain insights into the solution, we present some numerical results in Figure 7.1. We plot the deflection coefficient of the global test statistic as a function of attack parameters $P_i = P, \Delta_i = \Delta, \forall i$. We consider a 6-node network with the topology given by the undirected graph shown in Figure 2.1(c) deployed to detect a phenomenon. Nodes 1 and 2 are considered to be Byzantines. Sensing channel gains of the nodes are assumed to be $h = [0.8, 0.7, 0.72, 0.61, 0.69, 0.9]$ and weights are given by (7.2). We also assume that $M = 12$, $E_s = 5$, and $\sigma_i^2 = 1, \ \forall i$. Notice that, the deflection coefficient is zero when the condition in Lemma 7.4.1 is satisfied. Another observation to make is that the deflection coefficient can be made zero even when only two out of six nodes are Byzantines. Thus, by appropriately choosing attack parameters (P, Δ), less than 50% of data falsifying Byzantines can blind the network.

Next, using the probability of detection and the probability of false alarm as measures of detection performance, we investigate the degradation of transient detection performance of consensus algorithms with Byzantines. More specifically, we analyze the detection performance of the data

fusion scheme, denoted as $x(t+1) = W^t x(0)$, as a function of consensus iteration t by assuming that each node makes its local decision using the information available at the end of iteration t. For analytical tractability, we assume that $P_i = P, \ \forall i$. We denote by w_{ji}^t the element of matrix W^t in the jth row and ith column. Using these notations, we calculate the probability of detection and the probability of false alarm at the jth node at consensus iteration t. For clarity of exposition, we first derive our results for a small network with two Byzantine nodes and one honest node (see Appendix A.13). Due to the probabilistic nature of the Byzantine's behavior, it may behave as an honest node with a probability $(1-P)$. Let S denote the set of all combinations of such Byzantine strategies:

$$S = \{\{b_1, b_2\}, \{h_1, b_2\}, \{b_1, h_2\}, \{h_1, h_2\}\} \tag{7.6}$$

where by b_i we mean that Byzantine node i behaves as a Byzantine and by h_i we mean that Byzantine node i behaves as an honest node. Let $A_s \in U$ denote the indices of Byzantines behaving as an honest node in the strategy combination s, then, from (7.6) we have

$$U = \{A_1 = \{\}, A_2 = \{1\}, A_3 = \{2\}, A_4 = \{1, 2\}\}$$

$$U^c = \{A_1^c = \{1, 2\}, A_2^c = \{2\}, A_3^c = \{1\}, A_4^c = \{\}\}$$

where $\{\}$ is used to denote the null set. Let us use m_s to denote the cardinality of subset $A_s \in U$. Using these notations, we generalize our results for any arbitrary N.

Lemma 7.4.2. *The test statistic of node j at consensus iteration t, i.e., $\tilde{\Lambda}_j^t = \sum_{i=1}^{N_1} w_{ji}^t \tilde{Y}_i + \sum_{i=N_1+1}^{N} w_{ji}^t Y_i$ is a Gaussian mixture with PDF*

$$f(\tilde{\Lambda}_j^t | H_k) = \sum_{A_s \in U} P^{N_1 - m_s} (1-P)^{m_s} \phi\left((\mu_k)_{A_s} + \sum_{i=N_1+1}^{N} w_{ji}^t (\mu_{1k})_i, \sum_{i=1}^{N} (w_{ji}^t (\sigma_{1k})_i)^2 \right)$$

$$\textit{with } (\mu_k)_{A_s} = \sum_{u \in A_s} w_{ju}^t (\mu_{1k})_j + \sum_{u \in A_s^c} w_{ju}^t (\mu_{2k})_j.$$

The performance of the detection scheme in the presence of Byzantines can be represented in

terms of the probability of detection and the probability of false alarm of the network.

Proposition 7.4.3. *The probability of detection and the probability of false alarm of node j at consensus iteration t in the presence of Byzantines can be represented as*

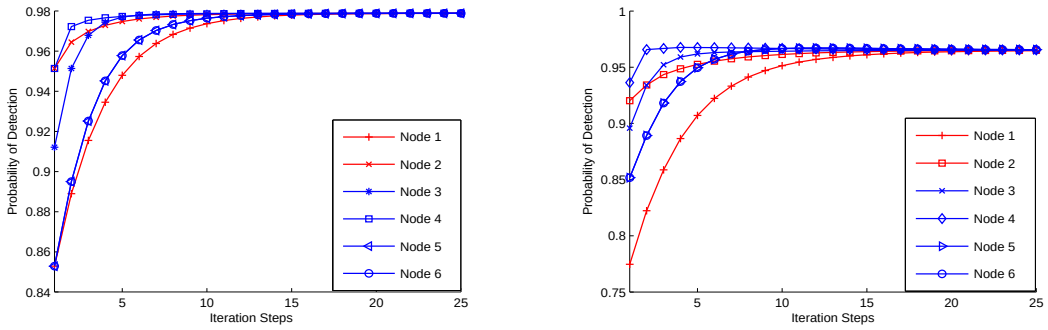
$$P_d^t(j) = \sum_{A_s \in U} P^{N_1 - m_s}(1-P)^{m_s} Q\left(\frac{\lambda - (\mu_1)_{A_s} - \sum_{i=N_1+1}^{N} w_{ji}^t (\mu_{11})_i}{\sqrt{\sum_{i=1}^{N} (w_{ji}^t (\sigma_{11})_i)^2}}\right) \text{ and}$$

$$P_f^t(j) = \sum_{A_s \in U} P^{N_1 - m_s}(1-P)^{m_s} Q\left(\frac{\lambda - (\mu_0)_{A_s} - \sum_{i=N_1+1}^{N} w_{ji}^t (\mu_{10})_i}{\sqrt{\sum_{i=1}^{N} (w_{ji}^t (\sigma_{10})_i)^2}}\right),$$

where λ is the threshold used for detection by node j.

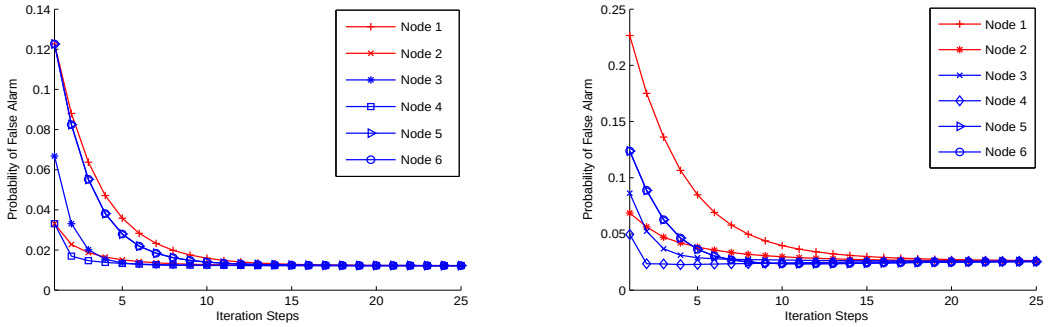
Next, to gain insights into the results given in Proposition 7.4.3, we present some numerical results in Figures 7.2 and 7.3. We consider the 6-node network shown in Figure 2.1(c) where the nodes employ the consensus algorithm 7.1 with $\epsilon = 0.6897$ to detect a phenomenon. Nodes 1 and 2 are considered to be Byzantines. We also assume that $\eta_i = 10$, $\sigma_i^2 = 2$, $\lambda = 33$ and $w_i = 1$. Attack parameters are assumed to be $(P_i, \Delta_i) = (0.5, 6)$ and $\tilde{w}_i = 1.1$. To characterize the transient performance of the weighted average consensus algorithm, in Figure 7.2(a), we plot the probability of detection as a function of the number of consensus iterations without Byzantines, i.e., $(\Delta_i = 0, \tilde{w}_i = w_i)$. Next, in Figure 7.2(b), we plot the probability of detection as a function of the number of consensus iterations in the presence of Byzantines. It can be seen that the detection performance degrades in the presence of Byzantines. In Figure 7.3(a), we plot the probability of false alarm as a function of the number of consensus iterations without Byzantines, i.e., $(\Delta_i = 0, \tilde{w}_i = w_i)$. Next, in Figure 7.3(b), we plot the probability of false alarm as a function of the number of consensus iterations in the presence of Byzantines. From both Figures 7.2 and 7.3, it can be seen that the Byzantine attack can severely degrade transient detection performance.

From the discussion in this section, we can see that Byzantines can severely degrade both the steady-state and the transient detection performance of conventional consensus-based detection algorithms. As mentioned earlier, a data falsifying Byzantine i can tamper its weight w_i as well as its sensing data Y_i to degrade detection performance. One approach to mitigate the effect of



(a) (b)

Fig. 7.2: (a) Probability of detection as a function of consensus iteration steps. (b) Probability of detection as a function of consensus iteration steps with Byzantines.



(a) (b)

Fig. 7.3: (a) Probability of false alarm as a function of consensus iteration steps. (b) Probability of false alarm as a function of consensus iteration steps with Byzantines.

sensing data falsification is to assign weights based on the quality of the data. In other words, a lower weight can be given to the data of the node identified as a Byzantine. However, to implement this approach one has to address the following two issues.

First, in the conventional weighted average consensus algorithm, weight w_i given to node i's data is controlled or updated by the node itself (see discussion in Section 7.3.1). Thus, a Byzantine node can always set a higher weight to its manipulated information and the final statistics will be dominated by the Byzantine nodes' local statistic that will lead to degraded detection performance. It will be impossible for any algorithm to detect this type of malicious behavior, since any weight that a Byzantine chooses for itself is a legitimate value that could also have been chosen by a node

that is functioning correctly. Thus, in the conventional consensus algorithms weight manipulation cannot be detected and therefore, conventional consensus algorithms cannot be used in the presence of an attacker.

Second, as will be seen later, the optimal weights given to nodes' sensing data depend on the following unknown parameters: identity of the nodes, which indicates whether the node is honest or Byzantine, and underlying statistical distribution of the nodes' data.

In the next section, we address these concerns by proposing a learning based robust weighted average consensus algorithm.

7.5 A Robust Consensus Based Detection Algorithm

In order to address the first issue discussed in Section 7.4, which is the optimal weight design, we propose a consensus algorithm in which the weight for node i's information is controlled (or updated) by the neighbors of node i rather than by node i itself. Note that, networks deploying such an algorithm are more robust to weight manipulation because if a Byzantine node j wants to assign an incorrect weight to the data of its neighbor i in the global test statistic, it has to ensure that all the neighbors of node i put the same incorrect weight as node j. Furthermore, the proposed algorithm enables the detection of weight-manipulating Byzantines (in contrast to conventional consensus algorithms). The attack can be treated as a consensus disruption attack and weight manipulation can be detected, unless all the neighbors of honest nodes are Byzantines.[3]

7.5.1 Distributed Algorithm for Weighted Average Consensus

In this section, we address the following questions: does there exist a distributed algorithm that solves the weighted average consensus problem while satisfying the condition that weights must be controlled (or updated) by neighbors $\mathcal{N}_i$ of node i rather than by node i itself? If such an algorithm exists, then, what are the conditions or constraints for the algorithm to converge?

[3]Weight manipulating Byzantines can be easily identified by techniques such as those given in [78, 94].

We consider a network of N nodes with a fixed and connected topology $G(V, E)$. Next, we state Perron-Frobenius theorem [38], which will be used later for the design and analysis of our robust weighted average consensus algorithm.

Theorem 7.5.1 ([38]). *Let W be a primitive non-negative matrix with left and right eigenvectors u and v, respectively, satisfying $Wv = v$ and $u^T W = u^T$. Then $\lim_{k\to\infty} W^k = \frac{vu^T}{v^T u}$.*

Using the above theorem, we take a reverse-engineering approach to design a modified Perron matrix $\hat{W}$ which has the weight vector $w = [w_1, w_2, \cdots, w_N]^T$, $w_i > 0$, $\forall i$ as its left eigenvector and $\vec{1}$ as its right eigenvector corresponding to eigenvalue 1. From the above theorem, if the modified Perron matrix $\hat{W}$ is primitive and non-negative, then a weighted average consensus can be achieved. Now, the problem boils down to designing such a $\hat{W}$ which meets our requirement that weights are controlled (or updated) by the neighbors $\mathcal{N}_i$ of node i rather than by node i itself.

For this purpose, we propose a modified Perron matrix $\hat{W} = I - \epsilon(T \otimes L)$ where L is the original graph Laplacian, $\otimes$ is element-wise matrix multiplication operator, and T is a transformation given by

$$[T]_{ij} = \begin{cases} \frac{\sum_{j\in\mathcal{N}_i} w_j}{l_{ii}} & \text{if } i = j \\ w_j & \text{otherwise.} \end{cases}$$

Observe that, the above transformation T satisfies the condition that weights are controlled (or updated) by neighbors $\mathcal{N}_i$ of node i rather than by node i itself. Based on the above transformation T, we propose our distributed consensus algorithm:

$$x_i(k+1) = x_i(k) + \epsilon \sum_{j\in\mathcal{N}_i} w_j(x_j(k) - x_i(k)).$$

Note that, the form of our update equation is different from the conventional update equation. Let us denote the modified Perron matrix by $\hat{W} = I - \epsilon\hat{L}$, where $\hat{L} = T \otimes L$.

We then explore the properties of the modified Perron matrix $\hat{W}$ and show that it satisfies the requirements of the Perron-Frobenius theorem [38]. These properties will later be utilized to prove the convergence of our proposed consensus algorithm.

Lemma 7.5.2. *Let G be a connected graph with N nodes. Then, the modified Perron matrix $\hat{W} = I - \epsilon(T \otimes L)$, with $0 < \epsilon < \frac{1}{\max_i \sum_{j \in \mathcal{N}_i} w_j}$ satisfies the following properties.*

1. *$\hat{W}$ is a nonnegative matrix with left eigenvector w and right eigenvector $\vec{1}$ corresponding to eigenvalue 1;*
2. *All eigenvalues of $\hat{W}$ are in a unit circle;*
3. *$\hat{W}$ is a primitive matrix*[4].

Proof. Notice that, $\hat{W}\vec{1} = \vec{1} - \epsilon(T \otimes L)\vec{1} = \vec{1}$ and $w^T\hat{W} = w^T - \epsilon w^T(T \otimes L) = w^T$. This implies that $\hat{W}$ has the left eigenvector w and the right eigenvector $\vec{1}$ corresponding to eigenvalue 1. To show that $\hat{W} = I + \epsilon T \otimes A - \epsilon T \otimes D$ is non-negative, it is sufficient to show that: $w > 0$, $\epsilon > 0$ and $\max_i \sum_{j \in \mathcal{N}_i} \epsilon\, w_j \leq 1, \forall i$. Since w is the left eigenvector of $\hat{L}$ and $w > 0$, $\hat{W}$ is non-negative if and only if

$$0 < \epsilon \leq \frac{1}{\max_i \sum_{j \in \mathcal{N}_i} w_j}.$$

To prove part $2)$, notice that all the eigenvectors of $\hat{W}$ and $\hat{L}$ are the same. Let γ_j be the jth eigenvalue of $\hat{L}$, then, the jth eigenvalue of $\hat{W}$ is $\lambda_j = 1 - \epsilon\gamma_j$. Now, part $2)$ can be proved by applying Gershgorin theorem [38] to the modified Laplacian matrix $\hat{L}$.

To prove part $3)$, note that G is strongly connected and, therefore, $\hat{W}$ is an irreducible matrix [38]. Thus, to prove that $\hat{W}$ is a primitive matrix, it is sufficient[5] to show that $\hat{W}$ has a single eigenvalue with maximum modulus of 1. In [75], the authors showed that when $0 < \epsilon < \max_i \sum_{j \neq i} a_{ij}$, the original Perron matrix W has only one eigenvalue with maximum modulus 1 at its spectral radius. Using a similar logic, $\hat{W}$ is a primitive matrix if

$$0 < \epsilon < \frac{1}{\max_i \sum_{j \in \mathcal{N}_i} w_j}.$$

[4] A matrix is primitive if it is non-negative and its mth power is positive for some natural number m.

[5] An irreducible stochastic matrix is primitive if it has only one eigenvalue with maximum modulus.

□

Theorem 7.5.3. *Consider a network with fixed and strongly connected undirected topology $G(V, E)$ that employs the distributed consensus algorithm*

$$x_i(k+1) = x_i(k) + \epsilon \sum_{j \in \mathcal{N}_i} w_j (x_j(k) - x_i(k))$$

where

$$0 < \epsilon < \frac{1}{\max\limits_i \sum\limits_{j \in \mathcal{N}_i} w_j}.$$

Then, consensus is reached asymptotically with $x^* = \frac{\sum_{i=1}^{N} w_i x_i(0)}{\sum_{i=1}^{n} w_i}, \forall i$.

Proof. A consensus is reached asymptotically, if the limit $\lim\limits_{k \to \infty} \hat{W}^k$ exists. According to Perron-Frobenius theorem [38], this limit exists for primitive matrices. Note that, $\vec{1} = [1, \cdots, 1]^T$ and w are right and left eigenvectors of the primitive nonnegative matrix $\hat{W}$ respectively. Thus, from [38]

$$\begin{aligned} x^* &= \lim_{k \to \infty} x(k) = \lim_{k \to \infty} (\hat{W})^k x(0) \\ x^* &= \vec{1} \frac{w^T x(0)}{w^T \vec{1}} \\ x^* &= \vec{1} \frac{\sum_{i=1}^{N} w_i x_i(0)}{\sum_{i=1}^{n} w_i} \end{aligned}$$

□

Next, to gain insights into the convergence property of the proposed algorithm, we present some numerical results in Figure 7.4. We consider the 6-node network shown in Figure 2.1(c) where the nodes employ the proposed algorithm (with $\epsilon = 0.3$) to reach a consensus. Next, we plot the updated state values at each node as a function of consensus iterations. We assume that the initial data vector is $x(0) = [5, 2, 7, 9, 8, 1]^T$ and the weight vector is $w = [0.65, 0.55, 0.48, 0.95, 0.93, 0.90]^T$. Note that, the parameter ϵ satisfies the condition mentioned in Theorem 7.5.3. Figure 7.4 shows the convergence of the proposed algorithm iterations. It is observed that within 20 iterations consensus has been reached on the global decision statistics, the weighted average of the initial values (states).

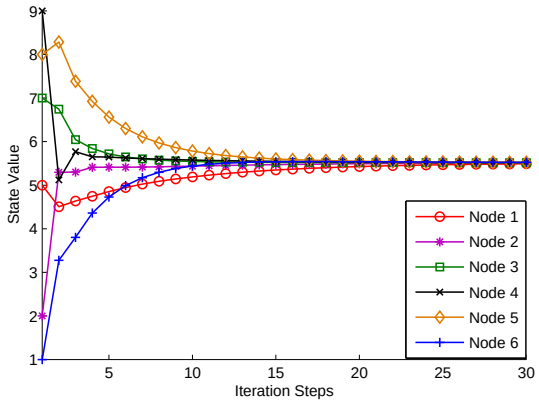


Fig. 7.4: Convergence of the network with a 6-nodes ($\epsilon = 0.3$).

In the proposed consensus algorithm, weights given to node i's data are updated by neighbors of the node i rather than by node i itself which addresses the first issue discussed in Section 7.4.

7.5.2 Adaptive Design of the Update Rules based on Learning of Nodes' Behavior

Next, to address the second issue discussed in Section 7.4, we exploit the statistical distribution of the sensing data and devise techniques to mitigate the influence of Byzantines on the distributed detection system. We propose a three-tier mitigation scheme where the following three steps are performed at each node: 1) identification of Byzantine neighbors, 2) estimation of parameters of identified Byzantine neighbors, and 3) adaptation of consensus algorithm (or update weights) using estimated parameters.

We first present the design of distributed optimal weights for the honest/Byzantine nodes assuming that the identities of the nodes are known. Later we will explain how the identity of nodes (i.e., honest/Byzantine) can be determined.

Design of Distributed Optimal Weights in the Presence of Byzantines

In this subsection, we derive closed form expressions for the distributed optimal weights which maximize the deflection coefficient. First, we consider the global test statistic $\Lambda = \frac{\sum_{i=1}^{N_1} w_i^B \tilde{Y}_i + \sum_{i=N_1+1}^{N} w_i^H Y_i}{\sum w}$

where $\sum w = \sum_{i=1}^{N_1} w_i^B + \sum_{i=N_1+1}^{N} w_i^H$ and obtain a closed form solution for optimal centralized weights. Then, we extend our analysis to the distributed scenario. Let us denote by δ_i^B, the centralized weight given to the Byzantine node and by δ_i^H, the centralized weight given to the Honest node. By considering $\delta_i^B = w_i^B / \sum w$ and $\delta_i^H = w_i^H / \sum w$, the optimal weight design problem can be stated formally as:

$$\max_{\{\delta_i^B\}_{i=1}^{N_1}, \{\delta_i^H\}_{i=N_1+1}^{N}} \frac{(\mu_1 - \mu_0)^2}{\sigma_{(0)}^2}$$
$$\text{s.t.} \quad \sum_{i=1}^{N_1} \delta_i^B + \sum_{i=N_1+1}^{N} \delta_i^H = 1$$

where μ_1, μ_0 and $\sigma_{(0)}^2$ are given in (7.3), (7.4) and (7.5), respectively. The solution of the above problem is presented in the next lemma.

Lemma 7.5.4. *Optimal centralized weights which maximize the deflection coefficient are given as*

$$\delta_i^B = \frac{w_i^B}{\sum_{i=1}^{N_1} w_i^B + \sum_{i=N_1+1}^{N} w_i^H},$$
$$\delta_i^H = \frac{w_i^H}{\sum_{i=1}^{N_1} w_i^B + \sum_{i=N_1+1}^{N} w_i^H}$$

where $w_i^B = \dfrac{(\eta_i \sigma_i^2 - 2P_i\Delta_i)}{\Delta_i^2 P_i(1-P_i) + 2M\sigma_i^4}$ *and* $w_i^H = \dfrac{\eta_i}{2M\sigma_i^2}$.

Proof. The above results can be obtained by setting the derivative of the deflection coefficient equal to zero and solving the equation. Note that, the constraint is trivially satisfied by normalizing the obtained weights as the value of deflection coefficient is unchanged after normalization. □

The optimality of the weights in Lemma 7.5.4 can also be verified by upper bounding the expression of deflection coefficient using the Cauchy-Schwarz inequality and observing that this upper bound is achieved by the weights in Lemma 7.5.4.

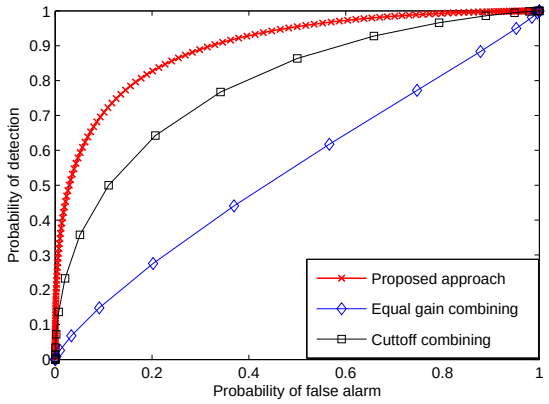


Fig. 7.5: ROC for different protection approaches

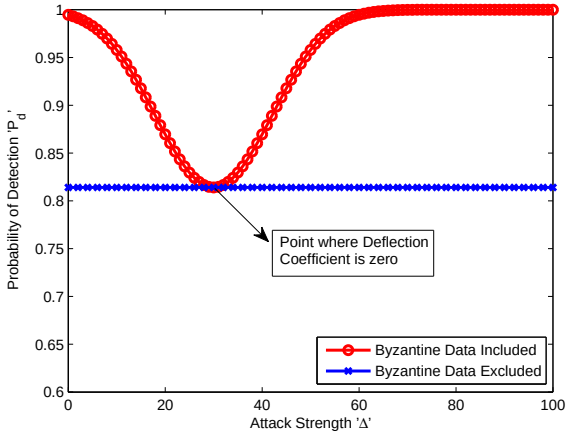


Fig. 7.6: Probability of Detection as a function of attack strength

Remark 1. *Distributed optimal weights can be chosen as* w_i^B *and* w_i^H*. Thus, the value of the global test statistic (or final weighted average consensus) is the same as the optimal centralized weighted combining scheme*[6].

Next, to gain insights into the solution, we present some numerical results in Figure 7.5 that corroborate our theoretical results. We assume that $M = 12$, $\eta_i = 3$, $\sigma_i^2 = 0.5$ and the attack parameters are $(P_i, \Delta_i) = (0.5, 9)$. In Figure 7.5, we compare our proposed weighted average

[6]Note that, weights w_i^B can be negative and in that case convergence of the proposed algorithm is not guaranteed. However, this situation can be dealt off-line by adding a constant value to make $w_i^B \geq 0$ and changing the threshold λ accordingly. More specifically, by choosing a constant c such that $\left(w_i^B + \frac{c}{x_i(0)}\right) \geq 0, \forall i$ and $\lambda \leftarrow \lambda + \beta c$ where β is the number of nodes with $w_i^B < 0$.

consensus-based detection scheme with the equal gain combining scheme[7] and the scheme where Byzantines are excluded from the fusion process. It can be clearly seen from the figure that our proposed scheme performs better than the rest of the schemes.

Next, in Figure 7.6, we plot the probability of detection of the proposed scheme as a function of the attack strength Δ for $P_i = 1$. The threshold λ is chosen to constrain the probability of false alarm below a constant $\delta = 0.01$. Also, we assume that $M = 12, \eta_i = 10, \sigma_i^2 = 6$. It can be seen from the figure that the worst detection performance is when the deflection coefficient is zero which also implies that the attacker is not providing any information or is excluded from the data fusion process. The reason for this is that when the deflection coefficient is non-zero, the information provided by the attacker is also non-zero which is being utilized by the proposed scheme to improve the detection performance.

Notice that, the optimal weights for the Byzantines are functions of the attack parameters $(P_i,\ \Delta_i)$, which may not be known to the neighboring nodes in practice. In addition, the parameters of the honest nodes might also not be known. Therefore, we propose a technique to learn or estimate these parameters. We then use these estimates to adaptively design the local fusion rule which are updated after each learning iteration.

Identification, Estimation, and Adaptive Fusion Rule

The first step at each node m is to determine the identity ($I^i \in \{H, B\}$) of its neighboring nodes $i \in \mathcal{N}_m$. Notice that, if node i is an honest node, its data under hypothesis H_k is normally distributed $\mathcal{N}((\mu_{1k})_i, (\sigma_{1k})_i^2)$. On the other hand, if node i is a Byzantine node, its data under hypothesis H_k is a Gaussian mixture which comes from $\mathcal{N}((\mu_{1k})_i, (\sigma_{1k})_i^2)$ with probability $(\alpha_1^i = 1 - P_i)$ and from $\mathcal{N}((\mu_{2k})_i, (\sigma_{2k})_i^2)$ with probability $\alpha_2^i = P_i$. Therefore, determining the identity ($I^i \in \{H, B\}$) of neighboring nodes $i \in \mathcal{N}_m$ can be posed as a hypothesis testing problem:

[7] In the equal gain combining scheme, all the nodes (including Byzantines) are given the same weight.

I_0 $(I^i = H)$: Y_i is generated from a Gaussian distribution under each hypothesis H_k;

I_1 $(I^i = B)$: Y_i is generated from a Gaussian mixture distribution under each hypothesis H_k.

Node classification can then be achieved using the maximum likelihood decision rule:

$$f(Y_i|\ I_0) \underset{B}{\overset{H}{\gtrless}} f(Y_i|\ I_1) \tag{7.7}$$

where $f(Y_i \mid I_l)$ is the probability density function (PDF) of Y_i under each hypothesis I_l. However, the parameters of the distributions are not known. Next, we propose a technique to learn these parameters. For an honest node i, the parameters to be estimated are $((\mu_{1k})_i, (\sigma_{1k})_i^2)$ and for Byzantines the unknown parameter set to be estimated is $\theta = \{\alpha_j^i, (\mu_{jk})_i, (\sigma_{jk})_i^2\}$, where $k = \{0, 1\}$, $j = \{1, 2\}$ and $i = 1, \cdots, N_m$, for N_m neighboring nodes. These parameters are estimated by observing the data over multiple learning iterations. In each learning iteration t, each node in the network employs the data coming from their neighbors for D detection intervals to learn their respective parameters. It is assumed that each node has the knowledge of the past D hypothesis test results (or history) through a feedback mechanism.[8] Also, notice that the learning is done in a separate learning phase and is not a part of consensus iterations.

First, we explain how the unknown parameter set for the distribution under the null hypothesis (I_0) can be estimated. Let us denote the data coming from an honest neighboring node i as $\mathbf{Y}_i(t) = [y_i^0(1), \cdots, y_i^0(D_1(t)), y_i^1(D_1(t)+1), \cdots, y_i^1(D)]$ where $D_1(t)$ denotes the number of times H_0 occurred in learning iteration t and y_i^k denotes the data of node i when the true hypothesis was H_k. To estimate the parameter set, $((\mu_{1k})_i, (\sigma_{1k})_i^2)$, of an honest neighboring node, one can employ a maximum likelihood based estimator (MLE). We use $((\hat{\mu}_{1k})_i(t), (\hat{\sigma}_{1k})_i^2(t))$ to denote the estimates at learning iteration t, where each learning iteration consists of D detection intervals. The ML

[8]Note that, there exist several applications where this assumption is valid, e.g., cognitive radio networks, sensor networks, etc. Also, the proposed method can be extended to the scenarios where the knowledge of the true hypothesis is not available at the cost of analytical tractability.

$$(\hat{\sigma}_{10})_i^2(t+1) = \frac{\sum_{r=1}^{t} D_1(r)[(\hat{\sigma}_{10})_i^2(t) + ((\hat{\mu}_{10})_i(t+1) - (\hat{\mu}_{10})_i(t))^2] + \sum_{d=1}^{D_1(t+1)} [y_i^0(d) - (\hat{\mu}_{10})_i(t+1)]^2}{\sum_{r=1}^{t+1} D_1(r)} \tag{7.8}$$

$$(\hat{\sigma}_{11})_i^2(t+1) = \frac{\sum_{r=1}^{t} (D - D_1(r))[(\hat{\sigma}_{11})_i^2(t) + ((\hat{\mu}_{11})_i(t+1) - (\hat{\mu}_{11})_i(t))^2] + \sum_{d=1}^{D-D_1(t+1)} [y_i^1(d) - (\hat{\mu}_{11})_i(t+1)]^2}{\sum_{r=1}^{t+1} (D - D_1(r))} \tag{7.9}$$

---

estimate of $((\mu_{1k})_i, (\sigma_{1k})_i^2)$ can be written in a recursive form:

$$(\hat{\mu}_{10})_i(t+1) = \frac{\sum_{r=1}^{t} D_1(r)}{\sum_{r=1}^{t+1} D_1(r)} (\hat{\mu}_{10})_i(t) + \frac{1}{\sum_{r=1}^{t+1} D_1(r)} \sum_{d=1}^{D_1(t+1)} y_i^0(d) \tag{7.10}$$

$$(\hat{\mu}_{11})_i(t+1) = \frac{\sum_{r=1}^{t} (D - D_1(r))}{\sum_{r=1}^{t+1} (D - D_1(r))} (\hat{\mu}_{11})_i(t) + \frac{1}{\sum_{r=1}^{t+1} (D - D_1(r))} \sum_{d=D_1(t+1)}^{D} y_i^1(d) \tag{7.11}$$

where expressions for $(\hat{\sigma_{10}})_i^2$ and $(\hat{\sigma_{11}})_i^2$ are given in (7.8) and (7.9), respectively. Observe that, by writing these expressions in a recursive manner, we need to store only D data samples at any given learning iteration t, but effectively use all tD data samples to determine the estimates.

Next, we explain how the unknown parameter set for the distribution under the alternate hypothesis (I_1) can be estimated. Since the data is distributed as a Gaussian mixture, we employ the expectation-maximization (EM) algorithm to estimate the unknown parameter set for Byzantines. Let us denote the data coming from a Byzantine neighbor i as $\tilde{\mathbf{Y}}_i(t) = [\tilde{y}_i^0(1), \cdots, \tilde{y}_i^0(D_1(t)), \tilde{y}_i^1(D_1(t)+1), \cdots, \tilde{y}_i^1(D)]$ where $D_1(t)$ denotes the number of times H_0 occurred in learning iteration t and $\tilde{y}_i^k$ denotes the data of node i when the true hypothesis was H_k. Let us denote the hidden variable as z_j with $j = \{1, 2\}$ or $(Z = [z_1, z_2])$. Now, the joint conditional PDF of $\tilde{y}_i^k$ and z_j, given the

parameter set, can be calculated to be

$$\begin{aligned} P(\tilde{y}_i^k(d), z_j|\theta) &= P(z_j|\tilde{y}_i^k(d), \theta) P(\tilde{y}_i^k(d)|(\mu_{jk})_i, (\sigma_{jk})_i^2) \\ &= \alpha_j^i P(\tilde{y}_i^k(d)|(\mu_{jk})_i, (\sigma_{jk})_i^2) \end{aligned}$$

In the expectation step of EM, we compute the expectation of the log-likelihood function with respect to the hidden variables z_j, given the measurements $\tilde{\mathbf{Y}}_i$, and the current estimate of the parameter set θ^l. This is given by

$$\begin{aligned} Q(\theta, \theta^l) &= E[\log P(\tilde{\mathbf{Y}}_i, Z|\theta)|\tilde{\mathbf{Y}}_i, \theta^l] \\ &= \sum_{j=1}^{2} \sum_{d=1}^{D_1(t)} \log[\alpha_j^i P(\tilde{y}_i^0(d)|(\mu_{j0})_i, (\sigma_{j0})_i^2) P(z_j|\tilde{y}_i^0(d), \theta^l)] \\ &+ \sum_{j=1}^{2} \sum_{d=D_1(t)+1}^{D} \log[\alpha_j^i P(\tilde{y}_i^1(d)|(\mu_{j1})_i, (\sigma_{j1})_i^2) P(z_j|\tilde{y}_i^1(d), \theta^l)] \end{aligned}$$

where

$$P(z_j|\tilde{y}_i^k(d), \theta^l) = \frac{\alpha_j^i(l) P(\tilde{y}_i^k(d)|(\mu_{jk})_i(l), (\sigma_{jk})_i^2(l))}{\sum_{n=1}^{2} \alpha_n^i(l) P(\tilde{y}_i^k(d)|(\mu_{nk})_i(l), (\sigma_{nk})_i^2(l))}. \tag{7.12}$$

In the maximization step of the EM algorithm, we maximize $Q(\theta, \theta^l)$ with respect to the parameter set θ so as to compute the next parameter set:

$$\theta^{l+1} = \arg\max_{\theta} \; Q(\theta, \theta^l).$$

First, we maximize $Q(\theta, \theta^l)$ subject to the constraint $\sum_{j=1}^{2} \alpha_j^i = 1$. We define the Lagrangian $\mathcal{L}$ as

$$\mathcal{L} = Q(\theta, \theta^l) + \lambda \left(\sum_{j=1}^{2} \alpha_j^i - 1 \right).$$

Now, we equate the derivative of $\mathcal{L}$ to zero:

$$\frac{d}{d\alpha_j^i}\mathcal{L} = \lambda + \frac{\sum_{d=1}^{D_1(t)} P(z_j|\tilde{y}_i^0(d), \theta^l)}{\alpha_j^i} + \frac{\sum_{d=D_1(t)+1}^{D} P(z_j|\tilde{y}_i^1(d), \theta^l)}{\alpha_j^i} = 0.$$

Multiplying both sides by α_j^i and summing over j gives $\lambda = -D$. Similarly, we equate the derivative of $Q(\theta, \theta^l)$ with respect to $(\mu_{jk})_i$ and $(\sigma_k)_i^2$ to zero. Now, an iterative algorithm for all the parameters is

$$\alpha_j^i(l+1) = \frac{1}{D}\left[\sum_{d=1}^{D_1(t)} P(z_j|\tilde{y}_i^0(d), \theta^l) + \sum_{d=D_1(t)+1}^{D} P(z_j|\tilde{y}_i^1(d), \theta^l) \right] \tag{7.13}$$

$$(\mu_{j0})_i(l+1) = \frac{\sum_{d=1}^{D_1(t)} P(z_j|\tilde{y}_i^0(d), \theta^l)\tilde{y}_i^0(d)}{\sum_{d=1}^{D_1(t)} P(z_j|\tilde{y}_i^0(d), \theta^l)} \tag{7.14}$$

$$(\mu_{j1})_i(l+1) = \frac{\sum_{d=D_1(t)+1}^{D} P(z_j|\tilde{y}_i^1(d), \theta^l)\tilde{y}_i^1(d)}{\sum_{d=D_1(t)+1}^{D} P(z_j|\tilde{y}_i^1(d), \theta^l)} \tag{7.15}$$

$$(\sigma_{j0})_i^2(l+1) = \frac{\sum_{j=1}^{2}\sum_{d=1}^{D_1(t)} P(z_j|\tilde{y}_i^0(d), \theta^l)(\tilde{y}_i^0(d) - (\mu_{j0})_i(l+1))^2}{\sum_{j=1}^{2}\sum_{d=1}^{D_1(t)} P(z_j|\tilde{y}_i^0(d), \theta^l)} \tag{7.16}$$

$$(\sigma_{j1})_i^2(l+1) = \frac{\sum_{j=1}^{2}\sum_{d=D_1(t)+1}^{D} P(z_j|\tilde{y}_i^1(d), \theta^l)(\tilde{y}_i^1(d) - (\mu_{j1})_i(l+1))^2}{\sum_{j=1}^{2}\sum_{d=D_1(t)+1}^{D} P(z_j|\tilde{y}_i^1(d), \theta^l)} \tag{7.17}$$

In the learning iteration t, let the estimates after the convergence of the above algorithm be denoted by $\hat{\theta}(t) = \{\hat{\alpha}_j^i(t), (\hat{\mu}_{jk})_i(t), (\hat{\sigma}_{jk})_i^2(t)\}$. These estimates are then used as the initial values for the next learning iteration $t+1$ that uses a new set of D data samples.

After learning the unknown parameter set under I_0 and I_1, node classification can be achieved using the following maximum likelihood decision rule:

$$\hat{f}(\mathbf{Y}_i|\ I_0) \quad \underset{B}{\overset{H}{\gtrless}} \quad \hat{f}(\mathbf{Y}_i|\ I_1) \tag{7.18}$$

where $\hat{f}(\cdot)$ is the PDF based on estimated parameters.

Using the above estimates and node classification, the optimal distributed weights for honest nodes after learning iteration t can be written as

$$w_i^H(t) = \frac{(\hat{\mu}_{11})_i(t) - (\hat{\mu}_{10})_i(t)}{(\hat{\sigma}_{10})_i^2(t)}. \tag{7.19}$$

Similarly, the optimal distributed weights for Byzantines after learning iteration t can be written as

$$w_i^B(t) = \frac{\sum_{j=1}^{2} \hat{\alpha}_j^i(t)[(\mu_{j1})_i(t) - (\hat{\mu}_{j0})_i(t)]}{\hat{\alpha}_1^i(t)\hat{\alpha}_2^i(t)\left((\hat{\mu}_{10}(t))_i - (\hat{\mu}_{20}(t))_i\right)^2 + \left(\hat{\alpha}_1^i(t)\left(\hat{\sigma}_{10}\right)_i^2(t) + \hat{\alpha}_2^i(t)\left(\hat{\sigma}_{20}\right)_i^2(t)\right)} \tag{7.20}$$

Next, we present some numerical results in Figure 7.7 to evaluate the performance of our proposed scheme. Consider the scenario where 6 nodes organized in an undirected graph (as shown in Figure 2.1(c)) are trying to detect a phenomenon. Node 1 and node 2 are considered to be Byzantines. We assume that $((\mu_{10})_i, (\sigma_{10})_i^2) = (3, 1.5)$, $((\mu_{11})_i, (\sigma_{11})_i^2) = (4, 2)$ and the attack parameters are $(P_i, \Delta_i) = (0.5, 9)$. In Figure 7.7, we plot ROC curves for different number of learning iterations. For every learning iteration, we assume that $D_1 = 10$ and $D = 20$. It can be seen from Figure 7.7 that within 4 learning iterations, detection performance of the learning based weighted gain combining scheme approaches the detection performance of weighted gain

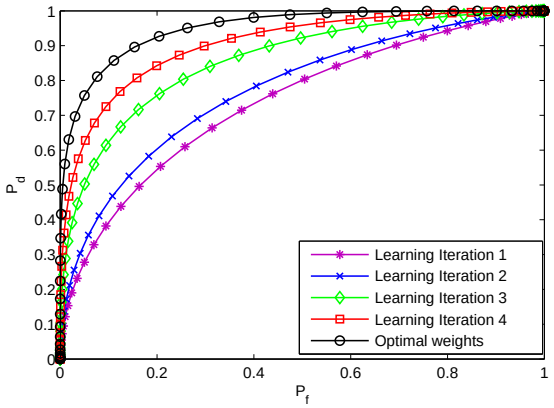


Fig. 7.7: ROC for different learning iterations

combining with known optimal weight based scheme.

Note that, the above learning based scheme can be used in conjunction with the proposed weighted average consensus-based algorithm to mitigate the effect of Byzantines.

7.6 Discussion

In this chapter, we analyzed the security performance of conventional consensus-based algorithms in the presence of data falsification attacks. We showed that above a certain fraction of Byzantine attackers in the network, existing consensus-based detection algorithm are ineffective. Next, we proposed a robust distributed weighted average consensus algorithm and devised a learning technique to estimate the operating parameters (or weights) of the nodes. This enables an adaptive design of the local fusion or update rules to mitigate the effect of data falsification attacks. We demonstrated that the proposed scheme, which uses the information of the identified Byzantines to network's benefit, outperforms exclusion based approaches where the only defense is to identify and exclude the attackers from the consensus process.

CHAPTER 8
COMPRESSIVE DETECTION WITH AN EAVESDROPPER: POSITIVE EFFECT OF CORRUPTED DATA IN SECRECY PERFORMANCE

8.1 Introduction

Previous chapters highlighted the negative effect of corrupted data or data falsification on the inference performance of the system. However, it is possible for a system designer to utilize the corrupted data for network's benefit. Motivated by this fact, in this chapter, we study the positive use of falsified data to improve the secrecy performance of a distributed inference system. In this chapter, we are interested in solving the problem of detecting a high dimensional signal based on compressed measurements. To solve an inference problem where some prior information about the signal is available, a customized measurement scheme could be implemented such that the optimal inference performance is achieved for the particular signal. As an example, for a signal detection problem where the signal of interest is known, the optimal design is the matched filter which is de-

pendent on the signal itself. However, it is possible that the signal that we wish to infer about may evolve over time. Thus, we are often interested in universal or agnostic design. A few attempts have been made in this direction to address problems of inference in the Compressive Signal Processing (CSP) literature recently [25, 36, 119]. CSP techniques are universal and agnostic to the signal structure and provide deterministic guarantees for a wide variety of signal classes.

The authors in [10, 25, 29] considered the deterministic signal detection problem in the compressed measurement domain where the performance limits of detection with compressed measurements were investigated. For signals that are not necessarily sparse, it was shown that a certain performance loss occurs due to compression when compared to the optimal test that acquires original measurements using the traditional measurement scheme. For stochastic signals, the compressive detection problem (i.e., detecting stochastic signals in the compressed measurement domain) was considered in [118, 119]. Both works focused only on compressive detection of 'zero-mean' stochastic signals based on observations corrupted by additive noise. Closed form expressions were derived for performance limits and performance loss due to compression was characterized analytically. A signal classification problem based on compressed measurements was considered in [26] where the authors developed a manifold based model for compressive classification. The authors in [35, 36] studied the performance of compressive sampling in detection and classification setups and introduced the generalized restricted isometry property that states that the angle between two vectors is preserved under random projections. Sparse event detection by sensor networks under a CS framework was considered in [68]. The problem of detection of spectral targets based on noisy incoherent projections was addressed in [55]. Schemes for the design and optimization of projection matrices for signal detection with compressed measurements have been proposed in [7, 114].

As mentioned earlier, CSP techniques are universal and agnostic to the signal structure and, therefore, are attractive in many practical applications. Despite its attractiveness to solve high dimensional inference problems, CSP suffers from a few major drawbacks which limit its applicability in practice. A CS based measurement scheme incurs a certain performance loss due to

compression when compared to the traditional measurement scheme while detecting non sparse signals. This can be seen as the price one pays for universality in terms of inference performance. In this chapter, we propose a collaborative compressive detection (CCD) framework to compensate for the performance loss due to compression. The CCD framework comprises of a group of spatially distributed nodes which acquire vector observations regarding the phenomenon of interest. Nodes send a compressed summary of their observations to the Fusion Center (FC) where a global decision is made. In this setup, we characterize the trade-off between dimensionality reduction in a universal CS based measurement scheme and the achievable performance. It is worthwhile to point out that, in contrast to [118, 119] where compressive detection of 'zero-mean' stochastic signals was considered, we study a more general problem where the stochastic signals can have 'non zero-mean'. Note that, some of the existing results can be seen as a special case of analytical results derived in this chapter. For both the cases, we show that for a fixed signal to noise ratio (SNR), if the number of collaborating nodes is greater than $(1/c)$, where $0 \leq c \leq 1$ is the compression ratio, the loss due to compression can be recovered.

In a CCD framework, the FC receives compressive observation vectors from the nodes and makes the global decision about the presence of the signal vector. The transmissions by the nodes, however, may be observed by an eavesdropper. The secrecy of a detection system against eavesdropping attacks is of utmost importance [21]. In a fundamental sense, there are two motives for any eavesdropper (Eve), namely *selfishness* and *maliciousness*, to compromise the secrecy of a given inference network. For instance, some of the nodes within a cognitive radio network (CRN) may selfishly take advantage of the FC's inferences and may compete against the CRN in using the primary user's channels without paying any participation costs to the network moderator. In another example, if the radar decisions are leaked to a malicious aircraft, the adversary aircraft can maliciously adapt its strategy against a given distributed radar network accordingly so as to remain invisible to the radar and in clandestine pursuit of its mission. Therefore, in the recent past, there has been a lot of interest in the research community in addressing eavesdropping attacks on inference networks. Recently, a few attempts have been made to address the problem of eavesdropping

threats on distributed detection networks [44]. However, a similar study in a CSP framework is missing from the literature.

Next, we investigate the CCD problem when the network operates in the presence of an eavesdropper who wants to discover the state of the nature being monitored by the system. While secrecy issues with CS based measurement schemes have been considered in [1, 82, 84], our work is considerably different. In contrast to [1, 82, 84], where performance limits of secrecy of CS based measurement schemes were analyzed (under different assumptions), we look at the problem from a practical perspective. We pursue a more active approach where the problem of optimal system design with secrecy guarantees is studied in an optimization setup. More specifically, we propose to use cooperating trustworthy nodes that assist the FC by injecting corrupted data to deceive the eavesdroppers to improve the secrecy performance of the system.[1] The addition of corrupted data to node transmissions is a data falsification scheme that is employed to mislead the eavesdropper. We consider the problem of determining optimal system parameters which maximize the detection performance at the FC, while ensuring perfect secrecy at the eavesdropper. In the process of determining optimal system parameters, we seek the answer to the question: Does compression help in improving the secrecy performance of the system? At first glance, it seems intuitive that compression should always improve the secrecy performance. However, we show that this argument is not necessarily true. In fact, secrecy performance of the system is independent of the compression ratio in the perfect secrecy regime.

The rest of the chapter is organized as follows. Section 8.2 presents the observation model and the problem formulation. In Section 8.3, performance of collaborative compression detection is analyzed for both deterministic and random signal cases. In Section 8.4, we investigate the problem where the network operates in the presence of an eavesdropper and propose corrupted data injection techniques to improve secrecy performance. In Section 8.5, we study the problem of determining optimal system parameters which maximize the detection performance at the FC, while ensuring perfect secrecy at the eavesdropper. Concluding remarks and possible future directions are given

[1] Artificial noise injection is a popular technique to guarantee secrecy in a wireless communication system [33, 72].

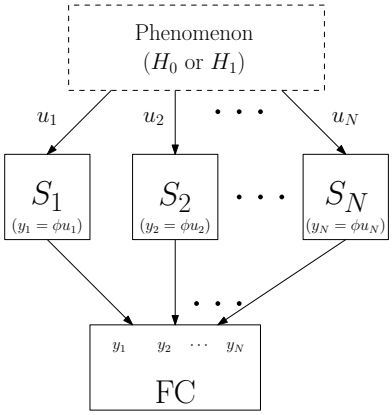


Fig. 8.1: Collaborative Compressive Detection Network

in Section 8.7.

8.2 Collaborative Compressive Detection

8.2.1 Observation Model

Consider two hypotheses H_0 (signal is absent) and H_1 (signal is present). Also, consider a parallel network, comprised of a central entity (known as the FC) and a set of N nodes, which faces the task of determining which of the two hypotheses is true (see Figure 8.1). Prior probabilities of the two hypotheses H_0 and H_1 are denoted by P_0 and P_1, respectively. The nodes observe the phenomenon (high dimensional signal), carry out local compression (low dimensional projection), and then send their local summary statistic to the FC. The FC makes a final decision after processing the locally compressed observations.

For the ith node observed signal, u_i can be modeled as

$$H_0 \ : \quad u_i = v_i$$

$$H_1 \ : \quad u_i = s + v_i$$

where u_i is the $P \times 1$ observation vector, s is either deterministic or random Gaussian signal vector (not necessarily sparse) to be detected. Specifically let $s \sim \mathcal{N}(\mu, \alpha^{-1} I_P)$ and additive noise $v_i \sim \mathcal{N}(0, \beta^{-1} I_P)$ where $x \sim \mathcal{N}(\mu, \Sigma)$ denotes that the vector x is distributed as multivariate Gaussian with mean vector μ and the covariance matrix Σ, and I_P is the $P \times P$ identity matrix. Note that, the deterministic signal can be considered as a special case of the random signal s with variance $\alpha^{-1} = 0$. Observations at the nodes are assumed to be conditionally independent and identically distributed.

Each node sends a M-length ($< P$) compressed version y_i of its P-length observation u_i to the FC. The collection of M-length universally sampled observations is given by, $y_i = \phi u_i$, where ϕ is an $M \times P$ projection matrix, which is assumed to be the same for all the nodes, and y_i is the $M \times 1$ compressed observation vector (local summary statistic).

Under the two hypotheses, the local summary statistic is

$$\begin{aligned} H_0 \ : \quad & y_i = \phi v_i \\ H_1 \ : \quad & y_i = \phi s + \phi v_i. \end{aligned}$$

The FC receives compressed observation vectors, $\mathbf{y} = [y_1, \cdots, y_N]$, from the nodes via error free communication channels and makes the global decision about the phenomenon.

8.2.2 Binary Hypothesis Testing at the Fusion Center

We consider the detection problem in a Bayesian setup where the performance criterion at the FC is the probability of error. The FC makes the global decision about the phenomenon by considering the likelihood ratio test (LRT) which is given by

$$\prod_{i=1}^{N} \frac{f_1(y_i)}{f_0(y_i)} \underset{H_0}{\overset{H_1}{\gtrless}} \frac{P_0}{P_1}. \tag{8.1}$$

Notice that, under the two hypotheses we have the following probability density functions (PDFs):

$$f_0(y_i) = \frac{\exp(-\frac{1}{2}y_i^T(\beta^{-1}\phi\phi^T)^{-1}y_i)}{|\beta^{-1}\phi\phi^T|^{1/2}(2\pi)^{M/2}}, \tag{8.2}$$

$$f_1(y_i) = \frac{\exp(-\frac{1}{2}(y_i-\phi\mu)^T((\alpha^{-1}+\beta^{-1})\phi\phi^T)^{-1}(y_i-\phi\mu))}{|(\alpha^{-1}+\beta^{-1})\phi\phi^T|^{1/2}(2\pi)^{M/2}}. \tag{8.3}$$

After plugging in (8.2) and (8.3) in (8.1) and taking logarithms on both sides, we obtain an equivalent test that simplifies to

$$\frac{\alpha^{-1}}{\beta^{-1}}\sum_{i=1}^{N} y_i^T(\phi\phi^T)^{-1}y_i + 2\sum_{i=1}^{N} y_i^T(\phi\phi^T)^{-1}\mu \underset{H_0}{\overset{H_1}{\gtrless}} \lambda$$

where $\lambda = (\alpha^{-1}+\beta^{-1})\left[2\log\frac{P_0}{P_1} + NM\log\left(1+\frac{\alpha^{-1}}{\beta^{-1}}\right)\right] + N(\phi\mu)^T(\phi\phi^T)^{-1}\phi\mu$.
For simplicity, we assume that $P_0 = P_1$. The test statistic for the collaborative compressive detector can be written in a compact form as

$$\Lambda(\mathbf{y}) = \frac{\alpha^{-1}}{\beta^{-1}}\sum_{i=1}^{N}\Lambda_1(y_i) + 2\sum_{i=1}^{N}\Lambda_2(y_i) \tag{8.4}$$

where $\Lambda_1(y_i) = y_i^T(\phi\phi^T)^{-1}y_i$ and $\Lambda_2(y_i) = y_i^T(\phi\phi^T)^{-1}\mu$.

We would like to point out that the test statistic for the deterministic signal and random signal with zero mean cases can be seen as a special case of the above test statistic. More specifically, for the deterministic signal s, the test statistic is given by $\Lambda(\mathbf{y}) = \sum_{i=1}^{N}\Lambda_1(y_i)$ and for the zero mean random signal the test statistic is given by $\Lambda(\mathbf{y}) = \sum_{i=1}^{N}\Lambda_2(y_i)$ which is consistent with [45] and [119].

8.3 Performance Analysis of Collaborative Compressive Detection

First, we look at the deterministic signal case and characterize the performance of the collaborative compressive detector.

8.3.1 Case I: Deterministic Signal

The optimal test at the FC can be written in a compact form as

$$\sum_{i=1}^{N} y_i^T(\phi\phi^T)^{-1}\phi s \quad \underset{H_0}{\overset{H_1}{\gtrless}} \quad \lambda,$$

with $\lambda = \frac{N}{2}s^T\phi^T(\phi\phi^T)^{-1}\phi s$. The decision statistic for the collaborative compressive detector is given as

$$\Lambda(\mathbf{y}) = \sum_{i=1}^{N} y_i^T(\phi\phi^T)^{-1}\phi s. \tag{8.5}$$

We analytically characterize the performance of the collaborative compressive detector in terms of the probability of error which is defined as

$$P_E = \frac{1}{2}P_F + \frac{1}{2}(1 - P_D)$$

where, $P_F = P(\Lambda(\mathbf{y}) > \lambda|H_0)$ and $P_D = P(\Lambda(\mathbf{y}) > \lambda|H_1)$ is the probability of false alarm and the probability of detection, respectively. To simplify the notations, we define

$$\hat{P} = \phi^T(\phi\phi^T)^{-1}\phi$$

as the orthogonal projection operator onto row space of ϕ. Using this notation, it is easy to show that

$$\Lambda(\mathbf{y}) \sim \begin{cases} \mathcal{N}(0, \beta^{-1}N\|\hat{P}s\|_2^2), & \text{under } H_0 \\ \mathcal{N}(N\|\hat{P}s\|_2^2, \beta^{-1}N\|\hat{P}s\|_2^2) & \text{under } H_1 \end{cases}$$

where $\|\hat{P}s\|_2^2 = s^T\phi^T(\phi\phi^T)^{-1}\phi s$.

Using techniques in [79], the probability of error can be calculated to be

$$P_E \;=\; Q\left(\frac{1}{2}\sqrt{\frac{N}{\beta^{-1}}}\|\hat{P}s\|_2\right) \tag{8.6}$$

where $Q(x) = \frac{1}{\sqrt{2\pi}}\int_x^\infty \exp(-\frac{u^2}{2})\,du$.

Next, we derive the modified deflection coefficient (first proposed in [81]) of the system and show its monotonic relationship with the probability of error as given in (8.6). The modified deflection coefficient provides a good measure of the detection performance since it characterizes the variance-normalized distance between the centers of two conditional PDFs. Notice that, for the deterministic signal case, y_i is distributed under the hypothesis H_j as, $y_i \sim \mathcal{N}(\mu_j^i, \Sigma_j^i)$. The modified deflection coefficient $D(\mathbf{y})$ can be obtained to be

$$\begin{aligned} D(\mathbf{y}) &= \sum_{i=1}^{N}(\mu_1^i - \mu_0^i)^T(\Sigma_1^i)^{-1}(\mu_1^i - \mu_0^i) \\ &= N\frac{\|\hat{P}s\|_2^2}{\beta^{-1}}. \end{aligned}$$

The monotonic relationship between P_E as given in (8.6) and $D(\mathbf{y})$ can be observed by noticing that

$$P_E = Q\left(\frac{\sqrt{D(\mathbf{y})}}{2}\right).$$

Later in the chapter, we will use the modified deflection coefficient to characterize the detection performance of the system.

Notice that, the detection performance is a function of the projection operator $\hat{P}$. In general, this performance could be either quite good or quite poor depending on the random projection matrix ϕ. Next, we provide bounds on the performance of the collaborative compressive detector

using the concept of ϵ-stable embedding.[2]

Definition 8.3.1. *Let $\epsilon \in (0,1)$ and $\mathcal{S}, \mathcal{X} \subset \mathbb{R}^P$. We say that a mapping ψ is an ϵ-stable embedding of $(\mathcal{S}, \mathcal{X})$ if*

$$(1-\epsilon)\,\|s-x\|_2^2 \leq \|\psi s - \psi x\|_2^2 \leq (1+\epsilon)\,\|s-x\|_2^2,$$

for all $s \in \mathcal{S}$ and $x \in \mathcal{X}$.

Using this concept, we state our result in the next theorem.

Theorem 8.3.2. *Suppose that $\sqrt{\frac{P}{M}}\hat{P}$ provides an ϵ-stable embedding of $(\mathcal{S}, \{0\})$. Then for any deterministic signal $s \in S$, the probability of error of the collaborative compressive detector satisfies*

$$Q\left(\sqrt{1+\epsilon}\frac{\sqrt{N}}{2}\sqrt{\frac{M}{P}}\frac{\|s\|_2}{\sqrt{\beta^{-1}}}\right) \leq P_E \leq Q\left(\sqrt{1-\epsilon}\frac{\sqrt{N}}{2}\sqrt{\frac{M}{P}}\frac{\|s\|_2}{\sqrt{\beta^{-1}}}\right).$$

Proof. By our assumption that $\sqrt{\frac{P}{M}}\hat{P}$ provides an ϵ-stable embedding of $(\mathcal{S}, \{0\})$, we know that

$$\sqrt{1-\epsilon}\,\|s\|_2 \leq \sqrt{\frac{P}{M}}\,\|\hat{P}s\|_2 \leq \sqrt{1+\epsilon}\,\|s\|_2. \tag{8.7}$$

Combining (8.7) with (8.6), the result follows. □

For small values of ϵ, P_E can be approximated as

$$P_E \approx Q\left(\frac{\sqrt{N}}{2}\sqrt{\frac{M}{P}}\frac{\|s\|_2}{\sqrt{\beta^{-1}}}\right).$$

The above expression tells us in a precise way how much information we lose by using low dimensional projections rather than the signal samples themselves. It also tells us how many nodes are needed to collaborate to compensate for the loss due to compression. More specifically, if $N \geq \frac{1}{c}$, where $c = \frac{M}{P}$ is defined as the compression ratio at each node, the loss due to compression

[2]To construct linear mappings that satisfy an ϵ-stable embedding property is beyond the scope of this work. We refer interested readers to [25].

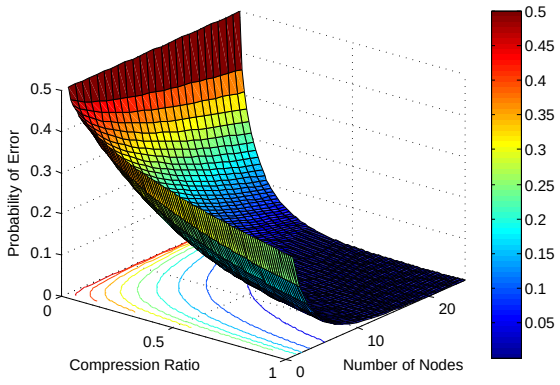


Fig. 8.2: Prob. of error as a function of number of nodes and compression ratio $c = M/P$ for $SNR = 3dB$

can be recovered. Notice that, for a fixed M, as the number of collaborating nodes approaches infinity, i.e., $N \to \infty$, the probability of error vanishes. On the other hand, to guarantee $P_E \leq \delta$, parameters $M,\ P$ and N should satisfy

$$cN \geq \frac{4}{SNR}(Q^{-1}(\delta))^2$$

where $SNR = \frac{\|s\|_2^2}{\beta^{-1}}$.

To corroborate our theoretical results, in Figure 8.2 we present the behavior of P_E with respect to collaboration and compression. We plot P_E as a function of the number of nodes N and compression ratio c. We assume that $SNR = 3dB$. It can be seen from Figure 8.2 that P_E is a monotonically decreasing function of c and N, and, therefore, the performance loss due to compression can be compensated by exploiting spatial diversity or collaboration.

In order to more clearly illustrate the behavior of P_E with respect to compression and collaboration, we also establish the following corollary of Theorem 8.3.2 using the Chernoff Bound.

Corollary 8.3.3. *Suppose that $\sqrt{\frac{P}{M}}\hat{P}$ provides an ϵ-stable embedding of $(\mathcal{S}, \{0\})$. Then for any deterministic signal $s \in S$, we have*

$$P_E \leq \frac{1}{2}\exp\left(-\frac{1}{8}cN\frac{\|s\|_2^2}{\beta^{-1}}\right).$$

Corollary 8.3.3 suggests that the error probability vanishes exponentially fast as we increase either the compression ratio c or the number of collaborating nodes N.

Next, we extend the above analysis to the case where the signal of interest is a random signal such that $s \sim \mathcal{N}(\mu, \alpha^{-1} I_P)$.

8.3.2 Case II: Random Signal with Arbitrary Mean

Let the signal of interest be $s \sim \mathcal{N}(\mu, \alpha^{-1} I_P)$ with an arbitrary μ. Then, the collaborative compressive detector is given by

$$\frac{\alpha^{-1}}{\beta^{-1}} \sum_{i=1}^{N} y_i^T (\phi\phi^T)^{-1} y_i + 2 \sum_{i=1}^{N} y_i^T (\phi\phi^T)^{-1} \mu \underset{H_0}{\overset{H_1}{\gtrless}} \lambda \tag{8.8}$$

where $\lambda = (\alpha^{-1} + \beta^{-1}) \left[NM \log \left(1 + \frac{\alpha^{-1}}{\beta^{-1}} \right) \right] + N(\phi\mu)^T (\phi\phi^T)^{-1} \phi\mu$. Note that, the test statistic is of the form $\sum_{i=1}^{N} \left[y_i^T A y_i + 2 b^T y_i \right]$ with $A = \frac{\alpha^{-1}}{\beta^{-1}} (\phi\phi^T)^{-1}$ and $b = (\phi\phi^T)^{-1} \phi\mu$. In general, it is difficult to find the PDF of such an expression in a closed form. It is worthwhile to point out that, in contrast to [118, 119] where compressive detection of 'zero-mean' stochastic signals was considered, we study a more general problem where the stochastic signals can have 'non zero-mean'. Using existing tools as given in [79], it is not possible to obtain a closed form performance analysis of the test (8.8).

Next, we state a Lemma from [74], which will be used to derive the distribution of the test statistic in a closed form.

Lemma 8.3.4 ([74]). *Let A be a symmetric matrix and $x \sim \mathcal{N}(\mu, V)$, where V is positive definite (hence nonsingular). The necessary and sufficient condition that $x^T A x + 2 b^T x + c$ follows a noncentral chi-squared distribution $\mathcal{X}_k^2(\delta)$ with k degrees of freedom and noncentrality parameter δ is that*

$$\begin{bmatrix} A \\ .. \\ b^T \end{bmatrix} V[A:b] = \begin{bmatrix} A & b \\ b^T & c \end{bmatrix} \tag{8.9}$$

in which case k is the rank of A and $\delta = \mu^T A\mu + 2b^T\mu + c$.

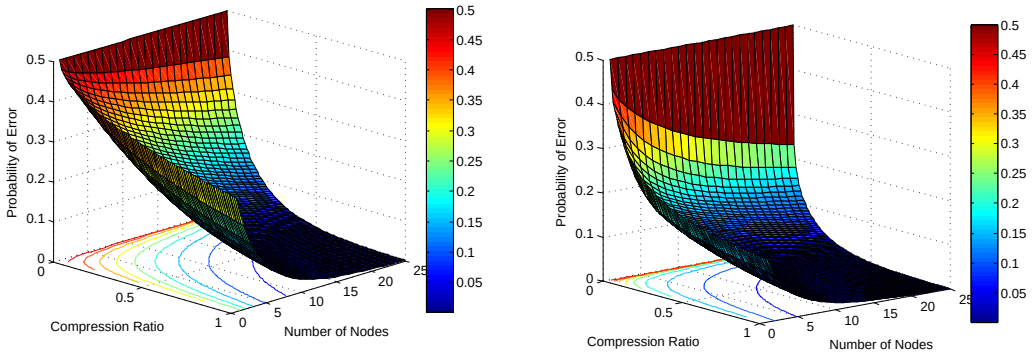
Next, using Lemma 8.3.4 we state the following proposition.

Proposition 8.3.5. *For a $P \times P$ symmetric and idempotent matrix S and $u_i \sim \mathcal{N}(\mu, \sigma^2 I_P)$, the test statistic of the form $u_i^T A u_i + 2b^T u_i + c$ with $A = \frac{1}{\sigma^2}S$, $b^T = \frac{1}{\sigma^2}z^T S$ and $c = \frac{1}{\sigma^2}z^T Sz$ follows a non-central chi-squared distribution $\mathcal{X}_k^2(\delta)$ where $k =$ Rank(S) and $\delta = \frac{1}{\sigma^2}\left(\mu^T S\mu + 2z^T S\mu + z^T Sz\right)$ for any arbitrary $P \times 1$ vector z.*

Proof. To prove the proposition, it is sufficient to show that the above mentioned A, b and c satisfy condition (8.9) in Lemma 8.3.4 for any arbitrary $P \times 1$ vector z. Notice that, S satisfies the following properties: symmetric $S^T = S$ and idempotent $S^2 = S$. Thus,

$$\begin{aligned}
\begin{bmatrix} A \\ .. \\ b^T \end{bmatrix} V[A:b] &= \begin{bmatrix} \frac{1}{\sigma^2}S \\ ..... \\ \frac{1}{\sigma^2}z^T S \end{bmatrix} \left[\sigma^2 I_P\right] \left[\frac{1}{\sigma^2}S \vdots \frac{1}{\sigma^2}Sz\right] \\
&= \begin{bmatrix} \hat{P} \\ .... \\ z^T S \end{bmatrix} \left[\frac{1}{\sigma^2}S \vdots \frac{1}{\sigma^2}Sz\right] \\
&= \begin{bmatrix} \frac{1}{\sigma^2}SS & \frac{1}{\sigma^2}SSz \\ \frac{1}{\sigma^2}z^T SS & \frac{1}{\sigma^2}z^T SSz \end{bmatrix} \\
&= \begin{bmatrix} A & b \\ b^T & c \end{bmatrix}
\end{aligned}$$

Thus, the test statistic follows a noncentral chi-squared distribution $\mathcal{X}_k^2(\delta)$ where $k =$ Rank(S) and $\delta = \frac{1}{\sigma^2}\left(\mu^T S\mu + 2z^T S\mu + z^T Sz\right)$ for any arbitrary z. □



(a) (b)

Fig. 8.3: Prob of error (P_e) analysis when $(\alpha^{-1}, \beta^{-1}) = (1, 20)$ and $P = 100$. (a) P_e with varying $((c,\ N))$ when $\mu = 0$. (b) P_e with varying $((c,\ N))$ when $\mu = 10^{-3}$.

Note that, the collaborative compressive detector is given by:

$$\frac{\alpha^{-1}}{\beta^{-1}} \sum_{i=1}^{N} y_i^T (\phi\phi^T)^{-1} y_i + 2 \sum_{i=1}^{N} y_i^T (\phi\phi^T)^{-1} \mu \underset{H_0}{\overset{H_1}{\gtrless}} \lambda.$$

Now, using the fact that $y_i = \phi u_i$ and rearranging the terms, we get

$$\sum_{i=1}^{N} \left[u_i^T \hat{P} u_i + 2 \frac{\beta^{-1}}{\alpha^{-1}} \mu^T \hat{P} u_i + \left(\frac{\beta^{-1}}{\alpha^{-1}} \right)^2 \mu^T \hat{P} \mu \right] \underset{H_0}{\overset{H_1}{\gtrless}} \tau$$

where $\hat{P} = \phi^T (\phi\phi^T)^{-1} \phi$ and $\tau = \frac{\beta^{-1}}{\alpha^{-1}} \lambda + N \left(\frac{\beta^{-1}}{\alpha^{-1}} \right)^2 \mu^T \hat{P} \mu$. Note that, the FC does not have access to u_i and the above test statistic is used only for deriving the PDF of the original test statistic.

Theorem 8.3.6. *For a projection matrix $\hat{P} = \phi^T (\phi\phi^T)^{-1} \phi$ and $u_i \sim \mathcal{N}(\mu, \sigma_k^2 I_P)$ under the hypothesis H_k, the test statistic*

$$\Lambda(\mathbf{y}) = \sum_{i=1}^{N} \left[u_i^T \hat{P} u_i + 2 \frac{\beta^{-1}}{\alpha^{-1}} \mu^T \hat{P} u_i + \left(\frac{\beta^{-1}}{\alpha^{-1}} \right)^2 \mu^T \hat{P} \mu \right]$$

has the following distribution

$$\frac{\Lambda(\mathbf{y})}{\sigma_k^2} \sim \begin{cases} \mathcal{X}^2_{NM}(\delta_0), & \textit{under } H_0 \\ \mathcal{X}^2_{NM}(N\delta_1) & \textit{under } H_1 \end{cases}$$

where $\sigma_0^2 = \beta^{-1}$, $\sigma_1^2 = \alpha^{-1} + \beta^{-1}$ *and* $\mathcal{X}^2_{NM}(\delta_k)$ *denotes noncentral chi-square distribution with* NM *degrees of freedom and parameters* $\delta_0 = 0$ *and* $\delta_1 = \frac{||\hat{P}\mu||_2^2}{\alpha^{-1}}\left(1 + \frac{\beta^{-1}}{\alpha^{-1}}\right)$.

Proof. Let us denote the test static by $\Lambda(\mathbf{y}) = \sum_{i=1}^{N} \Lambda(y_i)$ with

$$\Lambda(y_i) = \left[u_i^T \hat{P} u_i + 2\frac{\beta^{-1}}{\alpha^{-1}} \mu^T \hat{P} u_i + \left(\frac{\beta^{-1}}{\alpha^{-1}}\right)^2 \mu^T \hat{P} \mu \right].$$

Now notice that, $\frac{\Lambda(y_i)}{\sigma_k^2}$ is of the form $u_i^T A u_i + 2b^T u_i + c$ with $A = \frac{\hat{P}}{\sigma_k^2}$, $b^T = \frac{z^T \hat{P}}{\sigma_k^2}$, $z = \frac{\beta^{-1}}{\alpha^{-1}}\mu$ and $c = \frac{z^T \hat{P} z}{\sigma_k^2}$. Also note that, the projection matrix $\hat{P} = \phi^T(\phi\phi^T)^{-1}\phi$ is both symmetric and idempotent with rank$(\hat{P}) = M$. As a result, using Proposition 8.3.5 and the fact that $\frac{\Lambda(\mathbf{y})}{\sigma_k^2}$ is the sum of N I.I.D. chi-squared random variables $\frac{\Lambda(y_i)}{\sigma_k^2}$, the result in the Theorem 8.3.6 can be derived. □

If NM is large enough, using the central limit theorem, the following approximations hold

$$\frac{\Lambda(\mathbf{y})}{\sigma_k^2} \sim \begin{cases} \mathcal{N}(NM, 2NM), & \text{under } H_0 \\ \mathcal{N}((NM + N\delta_1), 2(NM + N\delta_1) & \text{under } H_1 \end{cases}$$

where $\delta_1 = \frac{||\hat{P}\mu||_2^2}{\alpha^{-1}}\left(1 + \frac{\beta^{-1}}{\alpha^{-1}}\right)$. As a result, we have

$$P_F = P\left(\frac{\Lambda(\mathbf{y})}{\beta^{-1}} > \frac{\tau}{\beta^{-1}} | H_0\right) = Q\left(\frac{\frac{\tau}{\beta^{-1}} - NM}{\sqrt{2NM}}\right)$$

and

$$\begin{aligned} P_D &= P\left(\frac{\Lambda(\mathbf{y})}{\alpha^{-1}+\beta^{-1}} > \frac{\tau}{\alpha^{-1}+\beta^{-1}}|H_1\right) \\ &= Q\left(\frac{\frac{\tau}{\alpha^{-1}+\beta^{-1}} - NM - N\delta_1}{\sqrt{2(NM+N\delta_1)}}\right) \end{aligned}$$

where $\tau = \frac{\beta^{-1}}{\alpha^{-1}}\lambda + N\left(\frac{\beta^{-1}}{\alpha^{-1}}\right)^2 ||\hat{P}\mu||_2^2$, $\lambda = (\alpha^{-1}+\beta^{-1})\left[NM\log\left(1+\frac{\alpha^{-1}}{\beta^{-1}}\right)\right] + N||\hat{P}\mu||_2^2$ and $\delta_1 = \frac{||\hat{P}\mu||_2^2}{\alpha^{-1}}\left(1+\frac{\beta^{-1}}{\alpha^{-1}}\right)$.

The detection performance of the system is a function of the projection operator $\hat{P}$. Next, we provide approximations to the performance of the collaborative compressive detector using the concept of ϵ-stable embedding of the mean μ.

Theorem 8.3.7. *Suppose that $\sqrt{\frac{P}{M}}\hat{P}$ provides an ϵ-stable embedding of $(\mathcal{U}, \{0\})$. Then for any random signal $s \sim \mathcal{N}(\mu, \alpha^{-1}I_P)$ with $\mu \in \mathcal{U}$, the probability of error of the collaborative compressive detector can be approximated as*

$$P_E = \frac{1}{2}Q\left(\sqrt{cN}\tau_0\right) + \frac{1}{2}\left(Q\left(\sqrt{cN}\tau_1\right)\right)$$

where $\tau_0 = \sqrt{\frac{P}{2}}\left((1+\tau^{-1})\left(\log(1+\tau) + \frac{\|\mu\|_2^2}{\alpha^{-1}P}\right) - 1\right)$, $\tau_1 = \sqrt{\frac{P+\delta_1'}{2}}\left(1 - \frac{\tau^{-1}\left(P\log(1+\tau)+\frac{\|\mu\|_2^2}{\alpha^{-1}}\right)}{P+\delta_1'}\right)$ with $\delta_1' = \frac{\|\mu\|_2^2}{\alpha^{-1}}(1+\tau^{-1})$ and $\tau = \frac{\alpha^{-1}}{\beta^{-1}}$.

Proof. By our assumption that $\sqrt{\frac{P}{M}}\hat{P}$ provides an ϵ-stable embedding of $(\mathcal{U}, \{0\})$, we know that

$$\sqrt{1-\epsilon}\,\|\mu\|_2 \leq \sqrt{\frac{P}{M}}\,\|\hat{P}\mu\|_2 \leq \sqrt{1+\epsilon}\,\|\mu\|_2. \tag{8.10}$$

In other words, for large values of NM the following approximation holds: $\|\hat{P}\mu\|_2^2 \approx \frac{M}{P}\|\mu\|_2^2 = c\|\mu\|_2^2$. The proof follows from the fact that $Q(x) = 1 - Q(-x)$ and by plugging in

$$P_F = Q\left(\sqrt{cN}\tau_0\right) \tag{8.11}$$

and

$$P_D = Q\left(-\sqrt{cN}\tau_1\right) \tag{8.12}$$

in the equation $P_E = \frac{1}{2}P_F + \frac{1}{2}(1 - P_D)$, the results stated in the theorem can be derived. □

Note that, $(-\tau_1) \leq \tau_0$ and, therefore, $P_D \geq P_F$. Similar to the deterministic signal case, if $N \geq c^{-1}$ the loss due to compression with a single node can be recovered in collaborative compressive detection for the random signal case as well. For a fixed M, as the number of collaborating nodes approaches infinity, i.e., $N \to \infty$, the probability of error vanishes. We would like to point out that by plugging in $\mu = 0$ in the above expressions, results for the zero mean signal case can be derived (which are consistent with [118, 119]).

To gain insights into Theorem 8.3.7, we present some illustrative examples that corroborate our results. In Figure 8.3(a) we plot the probability of error P_E as a function of the number of nodes N and compression ratio c. We assume that the signal of interest is $s \sim \mathcal{N}(0, I_P)$ and noise $v_i \sim \mathcal{N}(0, 20I_P)$, with the original length of the signal being $P = 100$. It can be seen from the figure that P_E is a monotonically decreasing function of (c, N). In Figure 8.3(b), we plot the probability of error P_E as a function of (c, N) when the signal of interest is $s \sim \mathcal{N}(\mu, I_P)$ with $\|\mu\|_2^2 = 10^{-3}$. Similar to Figure 8.3(a), P_E decreases monotonically with (c, N), however, with a much faster rate. In order to formally illustrate this behavior of P_E, we also establish the following corollary of Theorem 8.3.7.

Corollary 8.3.8. *Suppose that $\sqrt{\frac{P}{M}}\hat{P}$ provides an ϵ-stable embedding of $(\mathcal{U}, \{0\})$. Then for any random signal $s \sim \mathcal{N}(\mu, \alpha^{-1}I_P)$ with $\mu \in \mathcal{U}$, the error probability P_E of the collaborative compressive detector satisfies*

$$P_E \leq \frac{1}{4}\exp\left(-\frac{cN}{2}\tau_0^2\right) + \frac{1}{4}\exp\left(-\frac{cN}{2}\tau_1^2\right).$$

Proof. To prove the corollary, we first show that both τ_0 and τ_1 as given in Theorem 8.3.7 are positive. Let us denote by $\tau_k(\mu = 0)$ the expression when $\mu = 0$ is plugged in the expression for τ_k for $k \in \{0, 1\}$. Then, it can be shown that $\tau_k \geq \tau_k(\mu = 0)$ for $k \in \{0, 1\}$. Now, a sufficient

condition for $\tau_k > 0$ is $\tau_k(\mu = 0) > 0$ for $k \in \{0, 1\}$. The condition for $\tau_0(\mu = 0) > 0$ and $\tau_1(\mu = 0) > 0$ to be true can be written as

$$\frac{1}{1 + \frac{\beta^{-1}}{\alpha^{-1}}} < \log\left(1 + \frac{\alpha^{-1}}{\beta^{-1}}\right) < \frac{\alpha^{-1}}{\beta^{-1}}.$$

The above condition can be shown to be true by applying the logarithm inequality $\frac{\tau}{1+\tau} < \log(1 + \tau) < \tau$ with $\tau = \frac{\alpha^{-1}}{\beta^{-1}}$. Now using the Chernoff bound (i.e., $Q(x) \leq \frac{1}{2}\exp(-\frac{x^2}{2})$ for $x > 0$), it can be shown that

$$P_E \leq \frac{1}{4}\exp\left(-\frac{cN}{2}\tau_0^2\right) + \frac{1}{4}\exp\left(-\frac{cN}{2}\tau_1^2\right).$$

The above expression suggests that P_E vanishes exponentially fast as we increase either the compression ratio c or the number of collaborating nodes N. □

Next, we consider the problem where the network operates in the presence of an eavesdropper who wants to discover the state of the nature being monitored by the system. The FC's goal is to implement the appropriate countermeasures to keep the data regarding the presence of the phenomenon secret from the eavesdropper.

8.4 Collaborative Compressive Detection in the Presence of an Eavesdropper

In a collaborative compressive detection framework, the FC receives compressed observation vectors, $\mathbf{y} = [y_1, \cdots, y_N]$, from the nodes and makes the global decision about the presence of the random signal vector[3] $s \sim \mathcal{N}(\mu, \alpha^{-1}I_P)$ with $\mu \neq 0$. The transmissions of the nodes, however, may be observed by an eavesdropper who also wants to discover the state of the phenomenon (see Figure 8.4). To keep the data regarding the presence of the phenomenon secret from the eaves-

[3]In rest of the chapter, we will consider only the random signal detection case. Deterministic signal can be seen as a special case of random signal s with variance $\alpha^{-1} = 0$ and results for the deterministic signal case can be obtained by plugging in $\alpha^{-1} = 0$ in corresponding expressions for the random signal case.

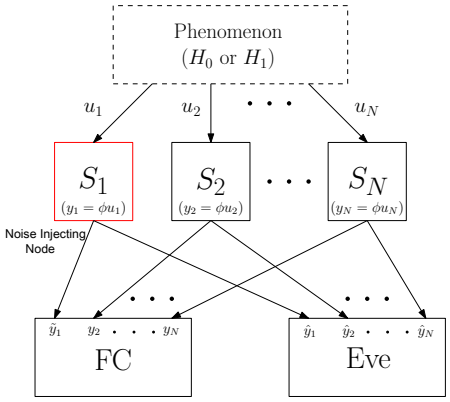


Fig. 8.4: Collaborative Compressive Detection Network in the Presence of an Eavesdropper

dropper, we propose to use cooperating trustworthy nodes that assist the FC by injecting corrupted data to mislead the eavesdroppers to improve the secrecy performance of the system.

8.4.1 Artificial Noise Injection Model

It is assumed that B out of N nodes (or α fraction of the nodes) inject corrupted data according to the model given next. Nodes tamper with their data y_i and send $\tilde{y}_i$ in the following manner:
Under H_0:

$$\tilde{y}_i = \begin{cases} \phi(v_i + W_i) & \text{with probability } P_1^0 \\ \phi(v_i - W_i) & \text{with probability } P_2^0 \\ \phi v_i & \text{with probability } (1 - P_1^0 - P_2^0) \end{cases}$$

Under H_1:

$$\tilde{y}_i = \begin{cases} \phi(s + v_i + W_i) & \text{with probability } P_1^1 \\ \phi(s + v_i - W_i) & \text{with probability } P_2^1 \\ \phi(s + v_i) & \text{with probability } (1 - P_1^1 - P_2^1) \end{cases}$$

where the signal s is assumed to be distributed as $s \sim \mathcal{N}(\mu, \alpha^{-1} I_P)$ and W_i is the corrupted data injected in the system which is distributed as AWGN $W_i \sim \mathcal{N}(D_i, \gamma^{-1} I_P)$ with $D_i = \kappa\mu$.

The parameter $\kappa > 0$ represents the corrupted data strength, which is zero for non corrupted data injecting nodes. Also note that, the values of (P_1^0, P_2^0) and (P_1^1, P_2^1) are system dependent. For example, under the assumption that the noise injecting nodes have perfect knowledge of the hypothesis, we have $P_1^0 = 1$ and $P_2^1 = 1$. In other scenarios, values of (P_1^0, P_2^0) and (P_1^1, P_2^1) are constrained by the local detection capability of the nodes. However, it is reasonable to assume that $(P_1^0 > P_2^0)$ and $(P_1^1 < P_2^1)$ because under hypothesis H_0 the tampered value should be high and under H_1 the tampered value should be low to degrade the performance at the eavesdropper. We assume that the observation model and corrupted data parameters (i.e., κ and γ^{-1}) are known to both the FC and the eavesdropper. The only information unavailable at the eavesdropper is the identity of the noise injecting nodes (Byzantines) and considers each node i to be Byzantine with a certain probability α.

8.4.2 Binary Hypothesis Testing in the Presence of an Eavesdropper

The FC can distinguish between y_i and $\tilde{y}_i$. Notice that, $\tilde{y}_i$ is distributed under the hypothesis H_0 as a multivariate Gaussian mixture $\mathcal{N}(P_k^0, \tilde{\mu_0}^i, \tilde{\Sigma_0}^i)$ which comes from $\mathcal{N}(\phi D_i, (\gamma^{-1} + \beta^{-1})\phi\phi^T)$ with probability P_1^0, from $\mathcal{N}(-\phi D_i, (\gamma^{-1} + \beta^{-1})\phi\phi^T)$ with probability P_2^0 and from $\mathcal{N}(0, (\gamma^{-1} + \beta^{-1})\phi\phi^T)$ with probability $(1 - P_1^0 - P_2^0)$. Similarly, under the hypothesis H_1 it is distributed as multivariate Gaussian mixture $\mathcal{N}(P_k^1, \tilde{\mu_1}^i, \tilde{\Sigma_1}^i)$ which comes from $\mathcal{N}(\phi(\mu + D_i), (\alpha^{-1} + \gamma^{-1} + \beta^{-1})\phi\phi^T)$ with probability P_1^1, from $\mathcal{N}(\phi(\mu - D_i), (\alpha^{-1} + \gamma^{-1} + \beta^{-1})\phi\phi^T)$ with probability P_2^1 and from $\mathcal{N}(\phi\mu, (\alpha^{-1} + \gamma^{-1} + \beta^{-1})\phi\phi^T)$ with probability $(1 - P_1^1 - P_2^1)$. The FC makes the global decision about the phenomenon by considering the likelihood ratio test (LRT) which is given by

$$\prod_{i=1}^{B} \frac{f_1(\tilde{y}_i)}{f_0(\tilde{y}_i)} \prod_{i=B+1}^{N} \frac{f_1(y_i)}{f_0(y_i)} \underset{H_0}{\overset{H_1}{\gtrless}} \frac{P_0}{P_1} \tag{8.13}$$

where $B/N = \alpha$. The eavesdropper is assumed to be unaware of the identity of the noise injecting Byzantines and considers each node i to be Byzantine with a certain probability α. Thus, the distribution of the data $\hat{y}_i$ at the eavesdropper under hypothesis H_j can be approximated as IID

multivariate Gaussian mixture with the same Gaussian parameters $\mathcal{N}(\tilde{\mu_j}^i, \tilde{\Sigma_j}^i)$ as above, however, with rescaled mixing probabilities $(\alpha P_1^j, \alpha P_2^j, 1-\alpha P_1^j - \alpha P_2^j)$. The eavesdropper makes the global decision about the phenomenon by considering the likelihood ratio test (LRT) which is given by

$$\prod_{i=1}^{N} \frac{f_1(\hat{y}_i)}{f_0(\hat{y}_i)} \underset{H_0}{\overset{H_1}{\gtrless}} \frac{P_0}{P_1}. \tag{8.14}$$

Analyzing the performance of the likelihood ratio detector in (8.13) and (8.14) in a closed form is difficult in general. Thus, we use the modified deflection coefficient [81] in lieu of the probability of error of the system. Deflection coefficient reflects the output signal to noise ratio and widely used as a surrogate for system performance while optimizing the performance of detection systems. As stated earlier, the modified deflection coefficient is defined as

$$D(y_i) \;\; = \;\; (\mu_1^i - \mu_0^i)^T (\Sigma_1^i)^{-1} (\mu_1^i - \mu_0^i)$$

where μ_j^i and Σ_j^i are the mean and the covariance matrix of y_i under the hypothesis H_j, respectively. Using these notations, the modified deflection coefficient at the FC can be written as

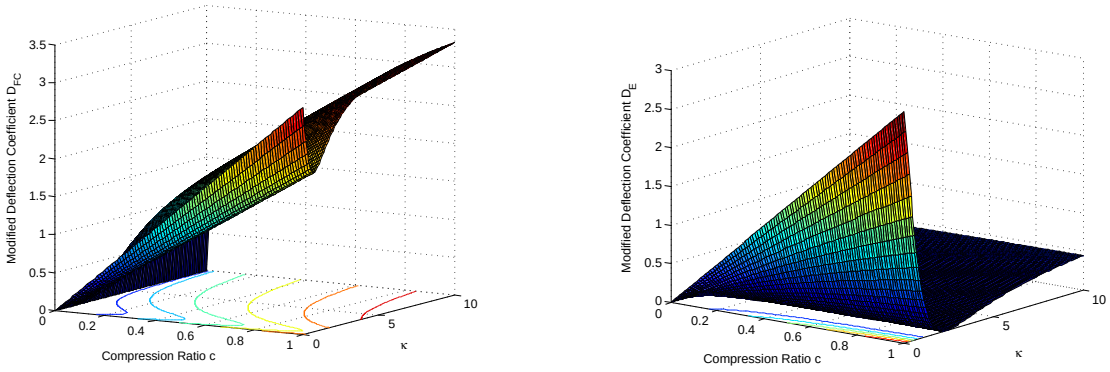
$$D(FC) = BD(\tilde{y_i}) + (N - B)D(y_i).$$

Dividing both sides of the above equation by N, we get

$$D_{FC} = \alpha D(\tilde{y_i}) + (1 - \alpha)D(y_i)$$

where $D_{FC} = \frac{D(FC)}{N}$ and will be used as the performance metric as a surrogate for the probability of error. Similarly, the modified deflection coefficient at the eavesdropper can be written as

$$D_{EV} = \frac{D(EV)}{N} = D(\hat{y}_i).$$



(a) (b)

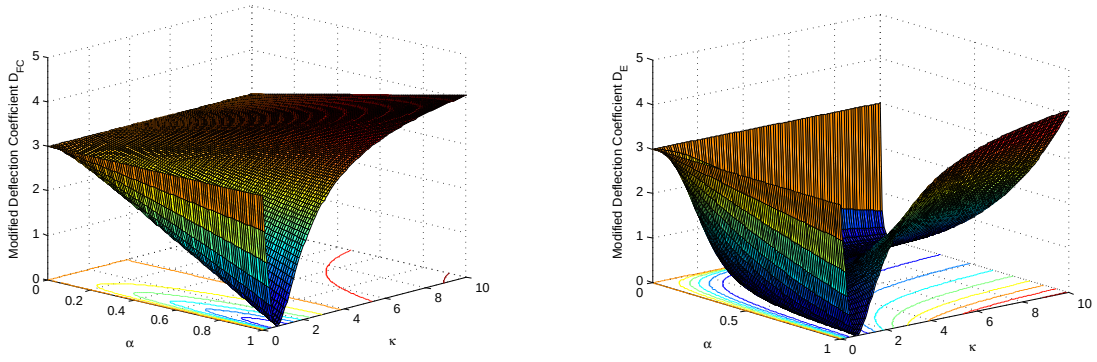
Fig. 8.5: Modified Deflection Coefficient analysis. (a) D_{FC} with varying c and κ. (b) D_{EV} with varying c and κ.

8.5 System Design with Physical Layer Secrecy Guarantees

Notice that, both D_{FC} and D_{EV} are functions of the compression ratio c and corrupted data injection parameters (α, W_i) which are under the control of the FC. This motivates us to obtain the optimal values of system parameters under a physical layer secrecy constraint. The problem can be formally stated as:

$$\begin{aligned} \underset{c,\alpha,W_i}{\text{maximize}} \quad & \alpha D(\tilde{y}_i) + (1-\alpha) D(y_i) \\ \text{subject to} \quad & D(\hat{y}_i) \leq \tau \end{aligned} \tag{8.15}$$

where $c = M/P$ is the compression ratio. We refer to $D(\hat{y}_i) \leq \tau$, where $\tau \geq 0$, as the physical layer secrecy constraint which reflects the secrecy performance of the system. The case where $\tau = 0$, or equivalently $D(\hat{y}_i) = 0$, is referred to as the perfect secrecy constraint. In the wiretap channel literature, it is typical to consider the maximum degree of information achieved by the main user (FC), while the information of the eavesdropper is exactly zero. This is commonly referred to as the perfect secrecy regime [120]. Next, we derive closed form expressions of the modified deflection coefficients at both the FC and the eavesdropper.



(a) (b)

Fig. 8.6: Modified Deflection Coefficient analysis. (a) D_{FC} with varying α and κ. (b) D_{EV} with varying α and κ.

8.5.1 Performance Analysis of Collaborative Compressive Detection with an Eavesdropper

A Closed Form Expression of the Modified Deflection Coefficient at the FC

As stated earlier, the modified deflection coefficient at the FC is

$$D_{FC} = \alpha D(\tilde{y}_i) + (1-\alpha)D(y_i).$$

Using (8.2) and (8.3), it can be shown that

$$D(y_i) = \frac{\|\hat{P}\mu\|_2^2}{\alpha^{-1} + \beta^{-1}},$$

where $\hat{P} = \phi^T(\phi\phi^T)\phi$.

Next, to derive $D(\tilde{y}_i)$, observe that $\tilde{y}_i$ is distributed as a multivariate Gaussian mixture with

$$\begin{aligned}
\tilde{\mu}_0^i &= (P_1^0 - P_2^0)\phi D_i \\
\tilde{\mu}_1^i &= (P_1^1 - P_2^1)\phi D_i + \phi\mu \\
\tilde{\Sigma}_1^i &= \sigma^2\phi\phi^T + \sum_{j=1}^{3} P_j^1(\tilde{\mu}_1^i(j) - \tilde{\mu}_1^i)(\tilde{\mu}_1^i(j) - \tilde{\mu}_1^i)^T
\end{aligned}$$

where $\tilde{\mu}_1^i(1) = \phi(\mu + D_i)$, $\tilde{\mu}_1^i(2) = \phi(\mu - D_i)$, $\tilde{\mu}_1^i(3) = \phi\mu$, $P_3^1 = 1 - P_1^1 - P_2^1$ and $\sigma^2 = (\alpha^{-1} + \beta^{-1} + \gamma^{-1})$. After some simplification, it can be shown that

$$\tilde{\Sigma}_1^i = \sigma^2\phi\phi^T + P_t[\phi D_i D_i^T \phi^T]$$

where $P_t = P_1^1 + P_2^1 - (P_1^1 - P_2^1)^2$. Also, notice that $\tilde{\Sigma}_1^i$ is of the form $A + bb^T$. Now, using the Sherman-Morrison formula [91], its inverse can be obtained to be

$$(\tilde{\Sigma}_1^i)^{-1} = \frac{(\phi\phi^T)^{-1}}{\sigma^2} - \frac{P_t(\phi\phi^T)^{-1}\phi D_i D_i^T \phi^T(\phi\phi^T)^{-1}}{\sigma^4 + \sigma^2 P_t D_i^T \phi^T(\phi\phi^T)^{-1}\phi D_i} \tag{8.16}$$

with $\sigma^2 = (\alpha^{-1} + \beta^{-1} + \gamma^{-1})$.

Also,

$$(\tilde{\mu}_1^i - \tilde{\mu}_0^i) = \phi\mu - P_b\phi D_i \tag{8.17}$$

where $P_b = (P_1^0 - P_2^0) + (P_2^1 - P_1^1)$. Using (8.16), (8.17) and the fact that $D_i = \kappa\mu$ where κ is referred to as the noise strength, the modified deflection coefficient $D(\tilde{y}_i)$ can be derived to be[4]

$$D(\tilde{y}_i) = (1 - P_b\kappa)^2 \frac{\|\hat{P}\mu\|_2^2}{\sigma^2} - P_t\kappa^2(1 - P_b\kappa)^2 \frac{\|\hat{P}\mu\|_2^4}{\sigma^2 r_b} \tag{8.18}$$

where $r_b = \sigma^2 + P_t\kappa^2\|\hat{P}\mu\|_2^2$ and $\sigma^2 = (\alpha^{-1} + \beta^{-1} + \gamma^{-1})$.

Proposition 8.5.1. *Suppose that $\sqrt{\frac{P}{M}}\hat{P}$ provides an ϵ-stable embedding of $(\mathcal{U}, \{0\})$. Then the modified deflection coefficient at the FC for any $\mu \in \mathcal{U}$ can be approximated as*

$$D_{FC} \approx \alpha \frac{(1 - P_b\kappa)^2}{\kappa^2 P_t + c^{-1}\frac{\sigma^2}{\|\mu\|_2^2}} + (1 - \alpha)c\frac{\|\mu\|_2^2}{\sigma^2 - \gamma^{-1}} \tag{8.19}$$

where

$P_b = (P_1^0 - P_2^0) + (P_2^1 - P_1^1)$, $P_t = P_1^1 + P_2^1 - (P_1^1 - P_2^1)^2$ *and* $\sigma^2 = (\alpha^{-1} + \beta^{-1} + \gamma^{-1})$.

Proof. Using the fact that $\sqrt{\frac{P}{M}}\hat{P}$ provides an ϵ-stable embedding of $(U, \{0\})$, for any $\mu \in \mathcal{U}$,

[4]For $\mu = 0$, $D_{FC} \approx \alpha\frac{P_b^2}{P_t + \frac{\sigma^2}{\|\hat{P}d\|}}$ where $D_i = d, \forall i$.

$D(y_i)$ and $D(\tilde{y}_i)$ can be approximated as

$$\begin{aligned} D(y_i) &= \frac{M}{P}\frac{\|\mu\|_2^2}{\sigma^2 - \gamma^{-1}} \\ D(\tilde{y}_i) &= \frac{M}{P}\frac{\|\mu\|_2^2}{\sigma^2}(1 - P_b\kappa)^2\left(1 - \frac{M}{P}\frac{\|\mu\|_2^2}{r_b}\kappa^2 P_t\right) \end{aligned}$$

where $r_b = \sigma^2 + P_t\kappa^2\|\hat{P}\mu\|_2^2$ and $\sigma^2 = (\alpha^{-1} + \beta^{-1} + \gamma^{-1})$. Plugging in the above values in $D_{FC} = \alpha D(\tilde{y}_i) + (1-\alpha)D(y_i)$ yields the desired result. □

A Closed Form Expression of the Modified Deflection Coefficients at the Eavesdropper

As stated earlier, the modified deflection coefficient of the eavesdropper is

$$D_{EV} = D(\hat{y}_i).$$

Next, to derive $D(\hat{y}_i)$, observe that $\hat{y}_i$ is distributed as a multivariate Gaussian mixture with

$$\begin{aligned} \hat{\mu}_0^i &= \alpha(P_1^0 - P_2^0)\phi D_i \\ \hat{\mu}_1^i &= \alpha(P_1^1 - P_2^1)\phi D_i + \phi\mu \\ \hat{\Sigma}_1^i &= \sigma^2\phi\phi^T + \sum_{j=1}^{3} p_j^1(\hat{\mu}_1^i(j) - \hat{\mu}_1^i)(\hat{\mu}_1^i(j) - \hat{\mu}_1^i)^T \end{aligned}$$

with $\hat{\mu}_1^i(1) = \phi(\mu + D_i)$, $\hat{\mu}_1^i(2) = \phi(\mu - D_i)$, $\hat{\mu}_1^i(3) = \phi\mu$, $p_1^1 = \alpha P_1^1$, $p_2^1 = \alpha P_2^1$, $p_3^1 = 1 - \alpha(P_1^1 - P_2^1)$ and $\sigma^2 = (\alpha^{-1} + \gamma^{-1} + \beta^{-1})$. Using these values, we state our result in the next proposition.[5]

Proposition 8.5.2. *Suppose that $\sqrt{\frac{P}{M}}\hat{P}$ provides an ϵ-stable embedding of $(\mathcal{U}, \{0\})$. Then, the modified deflection coefficient at the eavesdropper for any $\mu \in \mathcal{U}$ can be approximated as*

[5]For $\mu = 0$, $D_{EV} \approx \frac{(\alpha P_b)^2}{\alpha P_t^E + \frac{\sigma^2}{\|\hat{P}d\|}}$ where $D_i = d, \forall i$.

$$D_{EV} \approx \frac{(1-\alpha P_b \kappa)^2}{\alpha \kappa^2 P_t^E + c^{-1}\frac{\sigma^2}{\|\mu\|_2^2}} \tag{8.20}$$

where

$P_b = (P_1^0 - P_2^0) + (P_2^1 - P_1^1)$, $P_t^E = P_1^1 + P_2^1 - \alpha(P_1^1 - P_2^1)^2$ *and* $\sigma^2 = (\alpha^{-1} + \beta^{-1} + \gamma^{-1})$.

Proof. The proof is similar to that of Proposition 8.5.1 and is, therefore, omitted. □

In general, there is a trade-off between the detection performance and the secrecy performance of the system. To gain insights into this trade-off, in Figure 8.5 we plot the modified deflection coefficient, both at the FC and at the eavesdropper, as a function of compression ratio (c) and noise strength (κ) when $\alpha = 0.3,\ P_1^0 = P_2^1 = 0.8,\ P_2^0 = P_1^1 = 0.1$ and $\frac{\|\mu\|_2^2}{\sigma^2} = 3$. Next, in Figure 8.6 we plot the modified deflection coefficient, both at the FC and at the eavesdropper, as a function of the fraction of corrupted data injecting nodes (α) and noise strength (κ) when $P_1^0 = P_2^1 = 0.8,\ P_2^0 = P_1^1 = 0.1$ and $\frac{M}{P}\frac{\|\mu\|_2^2}{\sigma^2} = 3$. It can be seen from Figure 8.5 and Figure 8.6 that D_{FC} and D_{EV} do not exhibit nice properties (monotonicity or convexity) with respect to the system parameters and, therefore, it is not an easy task to design the system parameters under an arbitrary physical layer secrecy constraint. Also notice that, a specific case where the eavesdropper is completely blind deserves particular attention. This is referred to as the perfect secrecy regime, i.e., $D_{EV} = 0$. In the next subsection, we explore the problem of system design in a holistic manner in the perfect secrecy regime. More specifically, we are interested in analyzing the behavior of the modified deflection coefficient, both at the FC and at the eavesdropper, as a function of compression ratio $(c = M/P)$ and corrupted data injection parameters (α, W_i).

8.5.2 Optimal System Design Under Perfect Secrecy Constraint

The goal of the designer is to maximize the detection performance D_{FC}, while ensuring perfect secrecy at the eavesdropper, i.e., $\tau = 0$. The system design problem (8.15) under perfect secrecy constraint reduces to:

$$\begin{aligned} \underset{c,\alpha,W_i}{\text{maximize}} \quad & \alpha D(\tilde{y_i}) + (1-\alpha) D(y_i) \\ \text{subject to} \quad & D(\hat{y_i}) = 0 \end{aligned} \tag{8.21}$$

where c is the compression ratio and $(\alpha,\ W_i)$ are the corrupted data injection parameters. This reduction of the search space, which arises as a natural consequence of the perfect secrecy constraint, has the additional benefit of simplifying the mathematical analysis. Next, we explore the answer to the question: Does compression help in improving the secrecy performance of the system?

Does Compression Help?

We first consider the case where $\alpha P_b \kappa \neq 1$. In this regime, for fixed values of α, P_b and κ, the modified deflection coefficient, both at the FC and the eavesdropper, is a monotonically increasing function of the compression ratio. In other words, $\frac{dD_{FC}}{dc} > 0$ and $\frac{dD_{EV}}{dc} > 0$. This suggests that compression improves the secrecy performance at the expense of detection performance. More specifically, the FC would decrease the compression ratio until the physical layer secrecy constraint is satisfied. As a consequence, it will result in performance loss at the FC due to compression. In other words, there is a trade-off between the detection performance and the secrecy performance of the system. Observe that, $D_{EV} = 0$ if and only if $\alpha P_b \kappa = 1$ (ignoring the extreme conditions such as $c = 0$ or $\kappa = \infty$) and, in this regime, D_{FC} is a monotonically increasing function of the compression ratio c and D_{EV} is independent of the compression ratio c. These results are summarized in the the following proposition.

Proposition 8.5.3. *In the perfect secrecy regime* $(i.e., \alpha P_b \kappa = 1)$*,* D_{FC} *is a monotonically increasing function of the compression ratio* c *and* D_{EV} *is independent of the compression ratio* c*. When* $\alpha P_b \kappa \neq 1$*, the modified deflection coefficient, both at the FC and the eavesdropper, is a monotonically increasing function of the compression ratio.*

As mentioned above, in the perfect secrecy regime D_{EV} is independent of the compression ratio c and the network designer can fix $c = c_{max}$, where the value of c_{max} may be dependent

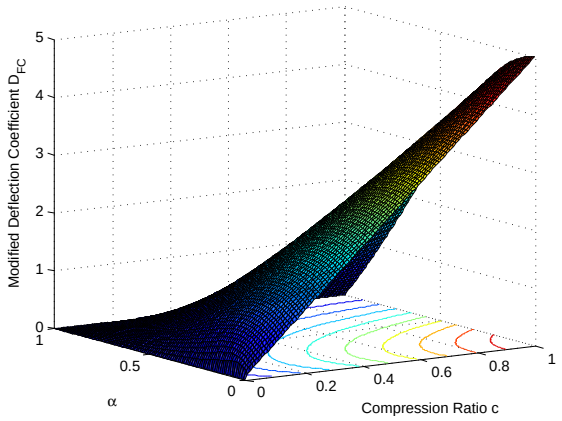


Fig. 8.7: Modified Deflection Coefficient as a function of α and compression ratio $c = M/P$ for $SNR = 5$dB in perfect secrecy regime.

on the application of interest. The system design problem under perfect secrecy constraint can be reformulated as

$$\arg\max_{\alpha, W_i} D_{FC}(c_{max}, \kappa = 1/(P_b\alpha)). \tag{8.22}$$

Next, we analyze the behavior of the $D_{FC}(c_{max}, \kappa = 1/(P_b\alpha))$ as a function of corrupted data injection parameters (α, W_i).

Optimal Artificial Noise Injection Parameters

Proposition 8.5.4. *In the high signal to noise ratio regime (defined as* $\frac{\|\mu\|_2^2}{\sigma^2} > \frac{P_b^2}{P_t}$ *where* $\sigma^2 = (\alpha^{-1} + \gamma^{-1} + \beta^{-1})$*), the modified deflection coefficient at the FC,* D_{FC}*, is a monotonically decreasing function of the fraction of data falsifying nodes (*$0 < \alpha \leq 1$*) under the perfect secrecy constraint.*

Proof. Deflection coefficient at the FC under the perfect secrecy constraint can be expressed as

$$D_{FC}(c_{max}, \kappa = 1/(P_b\alpha)) = \frac{\alpha(1-\frac{1}{\alpha})^2}{\frac{P_t}{\alpha^2 P_b^2} + \frac{1}{D}} + (1-\alpha)D + (1-\alpha)D_1$$

with $D = c_{max}\frac{\|\mu\|_2^2}{\sigma^2}$ and $D_1 = \dfrac{c_{max}\|\mu\|_2^2\gamma^{-1}}{\sigma^2(\alpha^{-1}+\beta^{-1})}$. Now, deriving the derivative of D_{FC} with respect

to α results in

$$\frac{dD_{FC}}{d\alpha} = \frac{P_t P_b^2 D^2(1-4\alpha+\alpha^2) - DP_b^4\alpha^2 - P_t^2 D^3}{(P_t D + \alpha^2 P_b^2)^2} - D_1.$$

Next, we show that $\frac{dD_{FC}}{d\alpha} < 0$.

First, let us define $F(\alpha) = x + \alpha^2\frac{1}{x} - (1+\alpha^2)$. It is easy to show that $F(\alpha)$ is a decreasing function of α if $x > 1$. This also implies that $F(\alpha) > 0$ if and only if $F(\alpha = 1) > 0 \Leftrightarrow (x+\frac{1}{x}) > 2$. Note that, $(x + \frac{1}{x}) > 2$, which follows from the fact that arithmetic mean is greater than the geometric mean. Having shown that $F(\alpha) > 0$, we return back to showing that $\frac{dD_{FC}}{d\alpha} < 0$. We start with the inequality

$$\begin{aligned} & x + \alpha^2\frac{1}{x} - (1+\alpha^2) > 0 \\ \Leftrightarrow \quad & x + \alpha^2\frac{1}{x} > (1+\alpha^2) \\ \Rightarrow \quad & \frac{4\alpha}{(1+\alpha^2)} + \frac{x}{(1+\alpha^2)} + \frac{\alpha^2}{(1+\alpha^2)}\frac{1}{x} > 1 \end{aligned}$$

Now, if we plug in $x = D\frac{P_t}{P_B^2}$ in the above inequality and rearrange the terms we get

$$\frac{P_t P_b^2 D^2(1-4\alpha+\alpha^2) - DP_b^4\alpha^2 - P_t^2 D^3}{(P_t D + \alpha^2 P_b^2)^2} < 0$$

which is true if $x = D\frac{P_t}{P_B^2} > 1$. □

Notice that, Proposition 8.5.3 suggests that to maximize the modified deflection coefficient D_{FC} under the perfect secrecy constraint (8.21), the network designer should choose the value of α as low as possible under the constraint that $\alpha > 0$ and accordingly increase κ to satisfy $\alpha P_b \kappa = 1$. In practice, α_{min} may be dependent on the application of interest.

Next, to gain insights into Propositions 8.5.3 and 8.5.4, we present some illustrative examples that corroborate our results. In Figure 8.7, we plot D_{FC} as a function of fraction of noise injection nodes α and compression ratio c in the perfect secrecy regime when $P_1^0 = P_2^1 = 0.8,\ P_2^0 = P_1^1 =$

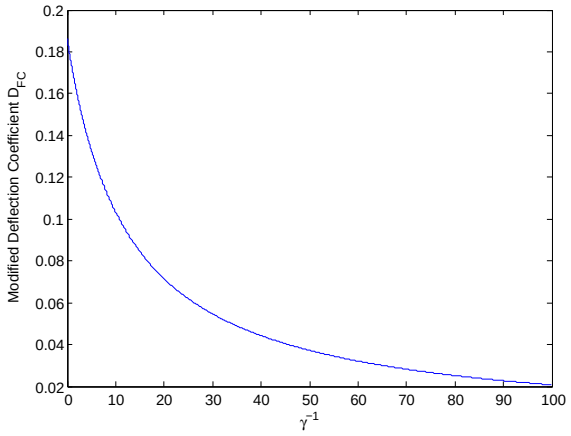


Fig. 8.8: Modified Deflection Coefficient as a function of γ^{-1} in perfect secrecy regime.

0.1 and $\frac{\|\mu\|_2^2}{\sigma^2} = 5dB$. It can be seen from the figure that D_{FC} is a monotonically increasing function of c and a monotonically decreasing function of α.

Next, we analyze the behavior of D_{FC} as a function of corrupted data variance γ^{-1}. This analysis will help us in determining the optimal corrupted data injection parameters.

Proposition 8.5.5. *In the high signal to noise ratio regime (defined as* $\frac{\|\mu\|_2^2}{\sigma^2} > \frac{P_b^2}{P_t}$ *where* $\sigma^2 = (\alpha^{-1} + \gamma^{-1} + \beta^{-1})$*) with perfect secrecy constraint* $(i.e.,\ \alpha P_b \kappa = 1)$*, the optimal corrupted data is a deterministic signal with value* $\frac{\mu}{\alpha_{min} P_b}$*, i.e.,* $f_{W_i}(w_i) = \delta(w_i - \frac{\mu}{\alpha_{min} P_b})$.

Proof. The proof follows from Proposition 8.5.4 and the fact that D_{FC} is a monotonically decreasing function of the variance γ^{-1} of the corrupted data, i.e., $\frac{dD_{FC}}{d\gamma^{-1}} < 0$. □

In Figure 8.8, we plot the modified deflection coefficient at the FC as a function of the variance of the corrupted data when $P_1^0 = P_2^1 = 0.8,\ P_2^0 = P_1^1 = 0.1$ and $(c, \alpha) = (0.2, 0.3)$. We assume that the signal of interest is $s \sim \mathcal{N}(\mu, I_P)$ with $\|\mu\|_2^2 = 5$ and noise $v_i \sim \mathcal{N}(0, 10 I_P)$. It can be seen that D_{FC} is a monotonically decreasing function of the corrupted data variance γ^{-1}. This observation implies that the optimal corrupted data is a deterministic signal. Using these results, the solution of the optimization problem (8.21) is summarized in the following theorem.

Theorem 8.5.6. *To maximize the modified deflection coefficient at the FC under the perfect secrecy constraint, the network designer should choose* $c = c_{max}$, $\alpha = \alpha_{min}$ *and deterministic corrupted data with value* $\frac{\mu}{\alpha_{min} P_b}$.

Notice that, Theorem 8.5.6 suggests that to maximize the modified deflection coefficient D_{FC} under the perfect secrecy constraint (8.21), the network designer should choose the value of α as low as possible under the constraint that $\alpha > 0$ and accordingly increase κ to satisfy $\alpha P_b \kappa = 1$. Also, the optimal corrupted data is a deterministic signal with value $\frac{\mu}{\alpha_{min} P_b}$, i.e., $f_{W_i}(w_i) = \delta(w_i - \frac{\mu}{\alpha_{min} P_b})$.

8.6 Measurement Matrix Design for Compressive Detection with Secrecy Guarantees

The random measurement scheme employed in CCD provides universality for a wide variety of signal classes, but it fails to exploit the signal structure that may be known *a priori*. To improve performance, optimization of the measurement scheme can be performed by exploiting the signal structure. In this section, we investigate the problem from a design perspective and consider the problem of measurement matrix design with secrecy guarantees in an optimization framework We show that the performance of the CCD framework can be significantly improved by using optimized measurement matrices (which exploit the underlying signal structure) along with corrupted data injection based techniques. More specifically, we design optimal measurement matrices which maximize the detection performance of the network while guaranteeing a certain level of secrecy considering three different scenarios: $1)$ signal of interest s is known, $2)$ s lies in low dimensional subspace, and $3)$ s is sparse.

8.6.1 Problem Formulation

We use the deflection coefficient as the detection performance metric in lieu of the probability of error of the system. Deflection coefficient reflects the output signal to noise ratio and is widely used in optimizing the performance of detection systems. The deflection coefficient at the ith node is defined as

$$D(y_i) \quad = \quad (\mu_1^i - \mu_0^i)^T (\Sigma_0^i)^{-1} (\mu_1^i - \mu_0^i)$$

where μ_j^i and Σ_j^i are the mean and the covariance matrix of y_i under the hypothesis H_j, respectively. Using these notations, the deflection coefficient at the FC can be written as $D(FC) = BD(\tilde{y_i}) + (N - B)D(y_i)$. Dividing both sides of the above equation by N, we get $D_{FC} = \alpha D(\tilde{y_i}) + (1 - \alpha)D(y_i)$ where $D_{FC} = D(FC)/N$ and will be used as the performance metric. Similarly, the deflection coefficient at the eavesdropper can be written as $D_{EV} = D(EV)/N = D(\hat{y_i})$. Notice that both D_{FC} and D_{EV} are functions of the measurement matrix ϕ and noise injection parameters (α, γ) which are under the control of the FC. This motivates us to design the optimal measurement matrix for fixed noise injection parameters (α, γ) under a physical layer secrecy constraint. The problem can be formally stated as:

$$\begin{aligned} \underset{\phi}{\text{maximize}} \quad & \alpha D(\tilde{y_i}) + (1 - \alpha) D(y_i) \\ \text{subject to} \quad & D(\hat{y_i}) \leq \tau \end{aligned} \tag{8.23}$$

where $\tau \geq 0$, is referred to as the physical layer secrecy constraint which reflects the security performance of the system. Earlier in the chapter, we have derived the expressions for D_{FC} and D_{EV}. Using those expressions (8.23) reduces to:

$$\begin{aligned} \underset{\phi}{\text{maximize}} \quad & \frac{\alpha(1 - P_b\gamma)^2}{\gamma^2 P_t + \frac{\sigma^2}{\|\hat{P}s\|_2^2}} + (1 - \alpha)\frac{\|\hat{P}s\|_2^2}{\sigma^2} \\ \text{subject to} \quad & \frac{(1 - \alpha P_b\gamma)^2}{\gamma^2 P_t^E + \frac{\sigma^2}{\|\hat{P}s\|_2^2}} \leq \tau \end{aligned} \tag{8.24}$$

where $\hat{P} = \phi^T(\phi\phi^T)^{-1}\phi$, $P_b = (P_1^0 - P_2^0) + (P_2^1 - P_1^1)$

$P_t = P_1^0 + P_2^0 - (P_1^0 - P_2^0)^2$ and

$P_t^E = \alpha(P_1^0 + P_2^0 - \alpha(P_1^0 - P_2^0)^2)$ matrix. Next, we solve (8.24) under various assumptions on the signal structure (e.g., known, low dimensional or sparse).

8.6.2 Optimal Measurement Matrix Design with Physical Layer Secrecy Guarantees

First, we explore some properties of the deflection coefficient at the FC, D_{FC}, and at the eavesdropper, D_{EV}, which will be used to simplify the measurement matrix design problem.

Proposition 8.6.1. *Deflection coefficient both at the FC and the Eve is a monotonically increasing function of* $D_H = \frac{\|\hat{P}s\|_2^2}{\sigma^2}$.

Proof. The proof follows from the fact that both $\frac{dD_{FC}}{dD_H} > 0$ and $\frac{dD_{EV}}{dD_H} > 0$. □

The above observation leads to the following equivalent optimal measurement matrix design problem for compressive detection:

$$\begin{aligned} \underset{\phi}{\text{maximize}} \quad & \delta = \|\hat{P}s\|_2^2 \\ \text{subject to} \quad & \|\hat{P}s\|_2^2 \leq \frac{\sigma^2}{\frac{(1-\alpha P_b\gamma)^2}{\tau} - \gamma^2 P_t^E} \end{aligned} \tag{8.25}$$

for any arbitrary signal s. Note that, for the random measurement matrix $\delta_r = \|\hat{P}s\|_2^2 = \frac{M}{N}\|s\|_2^2$ [25]. The factor M/N can be seen as the performance loss due to compression as the random measurement matrix fails to exploit the signal structure that may be known *a priori*. To improve performance, we consider the optimization of the measurement matrix by exploiting the signal structure while guaranteeing a certain level of secrecy. We show that any arbitrary secrecy constraint can be guaranteed by properly choosing the measurement matrix.

Known Signal Detection

First, we consider the case where s is known.

Lemma 8.6.2. *When s is known, the optimal value of the objective function of* (8.25)*, is given by* $\delta^* = \min\left(\|s\|_2^2, \frac{\sigma^2}{\frac{(1-\alpha P_b\gamma)^2}{\tau} - \gamma^2 P_t^E}\right)$.

Proof. The proof follows from the fact that $\hat{P}$ is an orthogonal projection operator, thus, $\|\hat{P}s\|_2^2 \leq \|s\|_2^2$. □

Let us denote the singular value decomposition of $\phi = U[\pi_M, 0]V^T$ where U is an $M \times M$ orthonormal matrix, $[\pi_M, 0]$ is an $M \times N$ diagonal matrix and V is an $N \times N$ orthonormal matrix. Now, the optimal ϕ which achieves δ^* is characterized in the following lemma.

Lemma 8.6.3. *When s is known, the optimal ϕ which achieves δ^* in* (8.25) *is given by $\phi^* = U[\pi_M, 0](V^*R)^T$ where U and diagonal $\pi_M > 0$ are totally arbitrary,*

$$R = \begin{bmatrix} \cos\theta & \mathbf{0} & \sin\theta \\ \mathbf{0} & \mathbf{I} & \mathbf{0} \\ -\sin\theta & \mathbf{0} & \cos\theta \end{bmatrix},$$

θ is the parameter which controls the level of secrecy such that $\theta = 0$ if $\|s\|_2^2 \leq \frac{\sigma^2}{\frac{(1-\alpha P_b\gamma)^2}{\tau} - \gamma^2 P_t^E}$, and, $\theta = \cos^{-1}\sqrt{\frac{\sigma^2/\|s\|_2^2}{\frac{(1-\alpha P_b\gamma)^2}{\tau} - \gamma^2 P_t^E}}$, otherwise. $V^* = [v_1^*, \cdots, v_N^*]$ is any orthonormal matrix satisfying $v_i^* \perp s, \forall i > M$.

Proof. To prove the lemma, notice that

$$\|\hat{P}s\|_2 = s^T V \begin{bmatrix} I_M & 0 \\ 0 & 0 \end{bmatrix} V^T s = \sum_{i=1}^{M} \tilde{s}_i^2 \leq \|s\|_2^2$$

where $\tilde{s} = V^T s$. The upper bound or equality in the above equation can be achieved if and only if $\tilde{s}_i = 0,\ \forall i > M$. The corresponding optimal measurement matrix for this case is characterized by $\phi^* = U[\pi_M, 0]V^T$ where the orthonormal U and diagonal $\pi_M > 0$ are totally arbitrary, while

$V = [v_1, \cdots, v_N]$, as seen above, has to be an orthonormal matrix satisfying $v_i \perp s, \forall i > M$. Now, the matrix (VR) is a orthonormal matrix for any orthonormal V and observe that the optimal V^* as given in the lemma is also orthonormal. Thus, for optimal ϕ^*, we have

$$\|\hat{P}s\|_2 = s^T V^* R \begin{bmatrix} I_M & 0 \\ 0 & 0 \end{bmatrix} (V^* R)^T s = \cos^2\theta \|s\|_2^2.$$

Next, using the definition of θ, the results in the lemma can be derived. □

If we define $\text{Proj}_u(w) = \frac{u^T w}{u^T u} u$ and $W = [w_1, \cdots, w_N]$ with $w_1 = s$ and w_k as any linearly independent set of vectors, one possible solution for V^* in a closed form is: $V^* = [v_1, \cdots, v_N]$, where $v_k = \frac{u_k}{\|u_k\|_2}$ and $u_k = w_k - \sum_{j=1}^{k-1} \text{Proj}_{u_j}(w_k)$. Note that, without physical layer secrecy constraint (or when $\theta = 0$) the optimal value of the objective function is $\|s\|_2^2$. Thus, there is no performance loss due to compression. With physical layer secrecy constraint, θ serves as a tuning parameter to guarantee a certain level of secrecy. This approach provides the optimal measurement matrix with a secrecy guarantee for a known s. However, in certain practical scenarios we do not have an exact knowledge of s. Next, we consider the cases where s is not completely known.

Low Dimensional Signal Detection

In this subsection, we consider the case where s is not completely known but is known to lie in a low dimensional subspace and design ϕ so that the detection performance at the FC is maximized while ensuring a certain level of secrecy at the eavesdropper. We assume that s resides in a K-dimensional subspace where $K < N$. That is to say, s can be expressed as $s = D\beta$ where D is an $N \times K$ matrix, whose columns are orthonormal, and β is the $K \times 1$ signal vector. Without loss of generality, we assume that $\|\beta\|_2^2 = 1$. Next, we look at the following two cases: $1)$ D can be designed, $2)$ D is fixed and known. For both the cases, we assume that β is deterministic but unknown and find ϕ which maximizes the worst case detection performance. Formally, for the case where D is a design parameter, the problem can be stated as

$$\begin{aligned} &\max_{\phi_{M\times N}} \max_{D_{N\times K}} \min_{\beta_{K\times 1}} \quad \delta = \|\hat{P}D\beta\|_2^2 \\ &\text{subject to} \qquad\qquad\quad \|\beta\|_2^2 = 1,\ \|\hat{P}D\beta\|_2^2 \leq \Delta \end{aligned} \tag{8.26}$$

where $\Delta = \frac{\sigma^2}{\frac{(1-\alpha P_b\gamma)^2}{\tau} - \gamma^2 P_t^E}$.

We state the Courant-Fischer theorem which will be used to solve the above optimization problem.

Theorem 8.6.4. *(Courant-Fischer [38]) Let A be a symmetric matrix with eigenvalues $\lambda_1 \geq \cdots \geq \lambda_N$ and S denote the any j-dimensional linear subspace of $\mathbb{C}^N$. Then,*

$$\max_{S:\, dim(S)=j} \min_{x\in S} \frac{x^T A x}{x^T x} = \lambda_j.$$

Lemma 8.6.5. *When s lies in a low dimensional signal subspace, the optimal value of the objective function of* (8.26) *is given by $\delta^* = \min\left(\|\beta\|_2^2, \Delta\right)$ if $K \leq M$, and $\delta^* = 0$, otherwise.*

Proof. Using Courant-Fischer theorem, we can show that the problem (8.26) without a physical layer secrecy constraint is equivalent to $\max_{\phi_{M\times N}} \lambda_k(\hat{P})$. Now, the proof follows by observing that $\hat{P}$ is the orthogonal projection operator and its eigenvalues are given by $\lambda_i = 1$ for $i = 1$ to M and $\lambda_i = 0$ for $i = M+1$ to N. □

Next, we assume that $K \leq M$ and characterize the optimal measurement matrix ϕ^* and the optimal subspace D^*.

Lemma 8.6.6. *When $s = D\beta$ and $K \leq M$, the optimal (ϕ^*, D^*) which achieves δ^* should satisfy the following condition: for any arbitrary $\phi = U[\pi_M, 0]V^T$ where $V = [v_1, \cdots, v_N]$, the optimal $D^* = \cos\theta D$ with $D = [v_1, \cdots, v_K]$.*

Proof. Note that for optimal (ϕ^*, D^*), we have

$$\begin{aligned}
\|\hat{P}s\|_2 &= \beta^T (D^*)^T V^* \begin{bmatrix} I_M & 0 \\ 0 & 0 \end{bmatrix} ((D^*)^T V^*)^T \beta \\
&= \beta^T \begin{bmatrix} \cos\theta I_K & 0 \end{bmatrix} \begin{bmatrix} I_M & 0 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} \cos\theta I_K \\ 0 \end{bmatrix} \beta \\
&= (\cos\theta)^2 \sum_{i=1}^{\min(K,M)} (\beta_i)^2
\end{aligned}$$

Observe $\min_{\beta} \sum_{i=1}^{\min(K,M)} (\beta_i)^2 = \|\beta\|_2^2$ if $\min(K, M) = K$ and 0, otherwise. Using the definition of θ, δ^* can be achieved. □

The above lemma can be interpreted as follows: for any fixed ϕ, one can choose D accordingly, so that the upper bound δ^* can be achieved. Next, we look at the case where D is fixed and we only optimize measurement matrix ϕ. Observe that,

$$\max_{\phi} \min_{\beta} \|\hat{P}D\beta\|_2^2 \leq \max_{\phi} \max_{D} \min_{\beta} \|\hat{P}D\beta\|_2^2 = \|\beta\|_2^2.$$

For a fixed D, the optimal value δ^* of the problem (8.26) serves as an upper bound. To simplify the problem, we introduce an $(N \times N)$ matrix P to guarantee secrecy in the system. In other words, $y_i = \phi P u_i$ where P is determined to guarantee physical layer secrecy. Next, we find ϕ for which this upper bound is achievable for a fixed D and P to secrecy.

Lemma 8.6.7. *For the low dimensional signal case $y_i = \phi P s$ with $s = D\beta$ where $D = [d_1, \cdots, d_K]$ is orthonormal, the optimal measurement matrix (ϕ^*, P^*), is given by $P^* = \cos\theta I_{N\times N}$ and $\phi^* = U[\pi_M, 0](V^*)^T$ where the orthonormal U and diagonal $\pi_M > 0$ are totally arbitrary, while $V^* = [v_1, \cdots, v_N]$ is such that $v_i = d_i$ for $i = 1$ to K and v_i for $i = K + 1$ to N are such that V forms an orthonormal basis.*

Proof. The proof is similar to Lemma 8.6.6, thus, omitted. □

For both the cases, where D can be designed and where D is fixed and known, without secrecy constraint the optimal value of the objective function is $\|s\|_2^2$. Thus, there is no performance loss due to compression. With secrecy constraint, θ serves as a tuning parameter to guarantee a certain level of secrecy.

Sparse Signal Detection

In this section, we assume that s is K-sparse in the standard canonical basis and $\|s\|_2^2 = 1$. Also, the exact number of the nonzero entries in s, their locations, and their values are assumed to be unknown. We design ϕ which maximizes the worst case detection performance by employing a lexicographic optimization approach[6]. Formally, the problem is

$$\begin{aligned} \max_{\phi_{M\times N}} \ \min_{s} \quad & \|\hat{P}s\|_2^2 \\ \text{subject to} \quad & \|s\|_2^2 = 1,\ \|s\|_0 = K, \\ & \|\hat{P}s\|_2^2 \leq \Delta,\ \phi \in \mathcal{A}_{K-1} \end{aligned} \tag{8.27}$$

where $\mathcal{A}_{K-1}$ is the set of solutions to the above optimization problem for sparsity level $K-1$ and Δ is defined in (8.26).

Lemma 8.6.8. *There is no performance loss while solving the problem* (8.27) *if we restrict our solution space to be matrices on the Stiefel manifold $S_t(M,N)$, where*

$$S_t(M,N) := \{\phi \in \mathbb{R}^{M\times N} : \phi\phi^T = I\}.$$

Proof. The proof follows from the observation that $\pi_M = I_M$ for frames $\phi = U[\pi_M, 0]V^T$ in Stiefel manifold and the value of $\|\hat{P}s\|_2^2$ is independent of π_M and U. □

Next, we limit our focus on Stiefel manifolds and establish an upper bound on the value of the objective function in (8.27) for different sparsity levels. Later we find measurement matrices which can achieve this upper bound.

[6]We first find a set of solutions that are optimal for a k_1-sparse signal. Then, within this set, we find a subset of solutions that are also optimal for (k_1+1)-sparse signals. This approach is known as a lexicographic optimization.

Lemma 8.6.9. *For the sparsity level $K = 1$, the optimal value of the objective function of* (8.27) *is* $\min\left(\frac{M}{N}, \Delta\right)$. *For the sparsity level $K \geq 2$, an upper bound on the value of the objective function is given by* $\min\left(\frac{M}{N}(1-\mu), \Delta\right)$, *where* $\mu = \sqrt{\frac{N-M}{M(N-1)}}$.

Proof. The proof is similar to Theorem 1 and Theorem 3 as given in [124], thus, omitted. □

Lemma 8.6.10. *The optimal measurement matrix (ϕ^*, P^*), for the K sparse signal case $y_i = \phi P s$ is given by:*

- *For the sparsity level $K = 1$, ϕ^* is a uniform tight frame with norm values equal to $\sqrt{M/N}$ and $P^* = \cos\theta I_{N\times N}$,*
- *For the sparsity level $K \geq 2$, ϕ^* is an equiangular tight frame with norm values equal to $\sqrt{M/N}$ and $P^* = \cos\theta I_{N\times N}$.*

Proof. Proof follows from the definition of uniform (or equiangular) tight frames [11] and observation that the upper bounds in the Lemma 8.6.9 can be reached only by these frames. □

Note that, without physical layer secrecy constraint (i.e., $\theta = 0$), our results reduce to the ones in [124]. With physical layer secrecy constraint, similar to previous cases, θ serves as a tuning parameter to guarantee an arbitrary level of secrecy. Also, it is shown that a real equiangular tight frame can exist only if $N \leq M(M+1)/2$, and a complex equiangular tight frame requires $N \leq M^2$ [101]. When M and N do not satisfy this condition, the bound in Lemma 8.6.9 can not be achieved and one can employ a heuristic or algorithmic approach [7].

8.7 Discussion

We considered the problem of collaborative compressive detection under a physical layer secrecy constraint. First, we proposed the collaborative compressive detection framework and showed that through collaboration the loss due to compression when using a single node can be recovered. Second, we studied the problem where the network operates in the presence of an eavesdropper.

We proposed the use of corrupted data injection techniques to improve secrecy performance. We also considered the problem of determining optimal system parameters which maximize the detection performance at the FC, while ensuring perfect secrecy at the eavesdropper. Optimal system parameters with perfect secrecy guarantees were obtained in a closed form. Finally, we designed optimal measurement matrices to obtain compressed data at distributed nodes so that the detection performance of the network is maximized while guaranteeing a certain level of secrecy. We solved the measurement matrix design problem for three different scenarios: a) the signal is known, b) the signal lies in a low dimensional subspace, and c) the signal is sparse. We showed that the secrecy performance of the system can be improved by using optimized measurement matrices along with corrupted data injection based techniques.

CHAPTER 9
CONCLUSION

9.1 Summary

In this thesis, the problem of accomplishing reliable inference from corrupted data was addressed. The general methodology for this was to first analyze the effect of data corrupting agents on inference for several practical network architectures and quantify their effect on the global performance of the network. It was found that as the network becomes more decentralized the susceptibility of the network to corrupted data increases. Moreover, in the case of Byzantine attacks, an adversary requires only few ($\leq 50\%$) malicious nodes to bring down the distributed inference system. The second step was to design schemes that are robust to such corrupted information from these agents in parallel, tree and peer to peer architectures. We followed the methodology suggested by Claude Shannon in his unpublished manuscript of 1956 titled "Reliable Machines from Unreliable Components" [89] which considers the problem of designing reliable machines from unreliable components. We employed three methods to improve system reliability: 1) improve individual system components, 2) use of error-correction codes, and 3) complete system redesign. These schemes used machine learning, game-theoretic and coding-theoretic approaches to improve the individual performance of the agents and/or correct the errors from them at the global agents. Specific contributions of this thesis are listed below.

In *Chapter 3*, we considered the problem of distributed Bayesian detection with Byzantine data, and characterized the power of attack analytically. For distributed detection under binary hypotheses, the expressions of the minimum attacking power to blind the FC was obtained. We showed that when there are more than 50% of Byzantines in the network, the data fusion scheme becomes blind and no detector can achieve any performance gain over the one based just on priors. The optimal attacking strategies for Byzantines that degrade the performance at the FC were obtained. Both, asymptotic and non-asymptotic cases were considered. It was shown that the results obtained for the non-asymptotic case are consistent with the results obtained for the asymptotic case only when the FC has the knowledge of the attacker's strategies, and thus, uses the optimal fusion rule. However, results obtained for the non asymptotic case, when the FC does not have knowledge of attacker's strategies, are not the same as the results obtained for the asymptotic case. It was also shown that the optimal attacking strategies in several cases have minimax property and, therefore, can be used to implement the optimal robust detector.

In *Chapter 4*, we considered the problem of distributed detection in perfect a-ary tree topologies in the presence of unlabeled Byzantine data, and characterized the power of attack analytically. We provided closed-form expressions for minimum attacking power required by the Byzantines to blind the FC. We obtained closed form expressions for the optimal attacking strategies that minimize the detection error exponent at the FC. We also looked at the possible counter-measures from the FC's perspective to protect the network from these Byzantines. We formulated the robust topology design problem as a bi-level program and provided an efficient algorithm to solve it.

In *Chapter 5*, we considered the problem of optimal Byzantine attacks on distributed detection mechanism in tree networks. We analyzed the performance limit of detection performance with Byzantines and obtained the optimal attacking strategies that minimize the detection error exponent. The problem was also studied from the network designer's perspective. It was shown that the optimal local detector is independent of the Byzantine's parameter. Next, we modeled the strategic interaction between the FC and the attacker as a Leader-Follower (Stackelberg) game and attacker and defender (FC) equilibrium strategies were identified. We also proposed a simple yet

efficient scheme to identify Byzantines and analytically evaluated its performance. There are still many interesting questions that remain to be explored in the future work such as analysis of the problem for arbitrary network topologies. The case where Byzantines collude in several groups (collaborate) to degrade the detection performance can also be investigated.

In *Chapter 6*, we considered the general framework of distributed inference problem in tree networks. We proposed an analytically tractable scheme to solve these problems and proved the asymptotic optimality of the proposed schemes. For the classification problem, when the number of hypotheses is $M = 2$, the proposed scheme is a majority-vote scheme for distributed detection in tree networks. Also, note that since the proposed scheme uses error-correcting codes, it works well even in scenarios with unreliable data. It should be pointed out that the proposed scheme is not limited to wireless sensor networks, although the application of wireless sensor networks has been considered in this paper. The DCFECC scheme has been found to be applicable to a number of other applications including the paradigm of crowdsourcing. We believe that one can use these results to address several other applications involving tree structures.

In *Chapter 7*, we analyzed the security performance of conventional consensus-based algorithms in the presence of data falsification attacks. We showed that above a certain fraction of Byzantine attackers in the network, existing consensus-based detection algorithm are ineffective. Next, we proposed a robust distributed weighted average consensus algorithm and devised a learning technique to estimate the operating parameters (or weights) of the nodes. This enables an adaptive design of the local fusion or update rules to mitigate the effect of data falsification attacks.

In *Chapter 8*, We considered the problem of collaborative compressive detection under a physical layer secrecy constraint. First, we proposed the collaborative compressive detection framework and showed that through collaboration the loss due to compression when using a single node can be recovered. Second, we studied the problem where the network operates in the presence of an eavesdropper. We proposed the use of artificial noise injection techniques to improve secrecy performance. We also considered the problem of determining optimal system parameters which

maximize the detection performance at the FC, while ensuring perfect secrecy at the eavesdropper. Optimal system parameters with perfect secrecy guarantees were obtained in a closed form. Further, we designed measurement matrices with secrecy guarantees for signal detection purposes. It was shown that the optimal design depends on the nature of the signal to be detected. Further, we showed that the detection performance of the system can be improved while guaranteeing a certain level of secrecy by using optimized measurement matrices along with friendly corrupted data injecting nodes.

9.2 Future Directions

There are a number of interesting future directions for research. Some specific future work that extends the work in different chapter is first discussed below.

In *Chapter 3*, the model where the Byzantines' sole aim is to disable the network and make the FC blind to the information sent by the local sensors was considered. This formulation results in a mathematical utility function which only contains the condition that approaches '0'. For this formulation, it has been found that the optimal attack for the Byzantines is to always flip their local result with probability '1'. One interesting problem is the analysis of 'Smart' Byzantines (or covert Byzantines [50]) which, besides aiming at disabling the network, also aim at protecting themselves from being detected. This analysis needs a mathematical formulation, where along with the utility function containing the 'blinding' aspect of Byzantines, there is an additional constraint defining the covertness of Byzantines from being identified. This would be an interesting problem as it is a more realistic scenario where malicious sensors would try to hide their malicious behavior.

In *Chapter 4* and *Chapter 5*, the problem of distributed detection in regular tree networks in the presence of Byzantines was considered. By modeling the strategic interaction between the FC and the attacker as a Leader-Follower (Stackelberg) game, attacker and defender (FC) equilibrium strategies were identified. There are still many interesting questions that remain to be explored in the future work such as utilizing more practical game theoretic models such as imperfect and

incomplete information games. Also, the analysis of the problem for non-regular topologies is worth exploring. The case where Byzantines collude in several groups (collaborate) to degrade the detection performance can also be investigated.

In *Chapter 6*, the use of error-correcting codes was considered for the distributed inference problem. However, some of the results were restrictive as they hold only under certain assumptions. In the future, one can extend this work by relaxing these assumptions. One can also extend this work to the case of target tracking when the target's location changes with time and the sensor network's aim is to track the target's motion. The proposed schemes provide an insight on M-ary search trees and show that the idea of coding-based schemes can also be used for other signal processing applications. For example, the application involving 'search' such as rumor source localization in social networks.

In *Chapter 7*, security performance of conventional consensus-based algorithms in the presence of data falsification attacks was analyzed. A robust distributed weighted average consensus algorithm and a learning technique to estimate the operating parameters (or weights) of the nodes were devised. This enabled an adaptive design of the local fusion or update rules to mitigate the effect of data falsification attacks. In the future, an analysis of the problem for time varying topologies can be done. Note that, some analytical methodologies used in this paper are certainly exploitable for studying the attacks in time varying topologies. Other questions such as the optimal topology so as to result in the fastest convergence rate and the problem with covert data falsification attacks with a smart adversary who disguises himself from the proposed detection scheme while accomplishing its attack can also be investigated. Also, in this thesis, we have assumed that the hypothesis does not change during the consensus iterations. One interesting direction to consider in the future is to study the problem where hypotheses are allowed to change during the information fusion phase.

In *Chapter 8*, we considered the problem of collaborative compressive detection under a physical layer secrecy constraint. We proposed the use of corrupted data injection techniques to improve secrecy performance. We also considered the problem of determining optimal system parameters which maximize the detection performance at the FC, while ensuring perfect secrecy at the eaves-

dropper. In the future work, an analysis of the problem in scenarios where the perfect secrecy constraint is relaxed is worth exploring. Note that, some analytical methodologies used in this paper are certainly exploitable for studying more general detection problems such as detection of non Gaussian signals in correlated noise. Other questions such as the case where communication channels are noisy can also be investigated.

Note that, high dimensional inference problem was motivated from the fact that the “big data" era requires a redesign of inference networks to handle high dimensional data. However, in most of the cases, while the data is of high-dimension, the information provided by them can be recovered efficiently from a low dimensional space. There is a need to redesign existing architectures of networks using concepts from low-dimensional signal processing while being robust to the external and internal attacks by malicious users such as the ones in [112]. Due to the presence of potential unreliable agents, one has to also take into consideration the robustness of the systems while developing such large-scale systems. This thesis demonstrated the utility of statistical learning techniques and tools from coding theory to achieve reliable performance from corrupted data.

APPENDIX A
APPENDIX

A.1 Proof of $0 \leq t^* \leq 1$

First, we show that $t^* \leq 1$. We start from the following equality:

$$\frac{\pi_{1,1}}{\pi_{1,0}} - 1 = \left(\frac{1-\pi_{1,0}}{\pi_{1,0}} - \frac{1-\pi_{1,1}}{\pi_{1,0}} \right) = \frac{1-\pi_{1,0}}{\pi_{1,0}} \left(1 - \frac{1-\pi_{1,1}}{1-\pi_{1,0}} \right). \tag{A.1}$$

By applying the logarithm inequality $1 - \frac{1}{x} < \ln(x) < (x-1),\ \forall x > 0$, to (A.1), we have

$$\begin{aligned} \ln \frac{\pi_{1,1}}{\pi_{1,0}} &< \frac{\pi_{1,1}}{\pi_{1,0}} - 1 \\ &= \frac{1-\pi_{1,0}}{\pi_{1,0}} \left(1 - \frac{1-\pi_{1,1}}{1-\pi_{1,0}} \right) \\ &\leq \frac{1-\pi_{1,0}}{\pi_{1,0}} \ln \frac{1-\pi_{1,0}}{1-\pi_{1,1}}. \end{aligned}$$

Now,

$$\begin{aligned}
& \ln\frac{\pi_{1,1}}{\pi_{1,0}} \le \frac{1-\pi_{1,0}}{\pi_{1,0}} \ln\frac{1-\pi_{1,0}}{1-\pi_{1,1}} \\
\Leftrightarrow \quad & \frac{\ln(\pi_{1,1}/\pi_{1,0})}{\ln((1-\pi_{1,0})/(1-\pi_{1,1}))} \frac{\pi_{1,1}}{1-\pi_{1,1}} \le \frac{(1/\pi_{1,0})-1}{(1/\pi_{1,1})-1} \\
\Leftrightarrow \quad & \frac{\ln\left(\frac{\ln(\pi_{1,1}/\pi_{1,0})}{\ln((1-\pi_{1,0})/(1-\pi_{1,1}))} \frac{\pi_{1,1}}{1-\pi_{1,1}}\right)}{\ln\left(\frac{(1/\pi_{1,0})-1}{(1/\pi_{1,1})-1}\right)} \le 1 \\
\Leftrightarrow \quad & t^* \le 1.
\end{aligned}$$

Next, we show that $t^* \ge 0$. First we prove that the denominator of t^* is nonnegative. Since $\pi_{1,1} > \pi_{1,0}$ for $P_d > P_f$ and $\alpha < 0.5$, we have

$$\pi_{1,1} > \pi_{1,0} \tag{A.2}$$

$$\Leftrightarrow \quad \frac{(1/\pi_{1,0})-1}{(1/\pi_{1,1})-1} \ge 1 \tag{A.3}$$

$$\Leftrightarrow \quad \ln\left(\frac{(1/\pi_{1,0})-1}{(1/\pi_{1,1})-1}\right) \ge 0. \tag{A.4}$$

Next we prove that the numerator of t^* is also nonnegative, and then t^* is nonnegative. We start from the following equality:

$$1-\frac{\pi_{1,0}}{\pi_{1,1}} = \left(\frac{1-\pi_{1,0}}{\pi_{1,1}} - \frac{1-\pi_{1,1}}{\pi_{1,1}}\right) = \frac{1-\pi_{1,1}}{\pi_{1,1}}\left(\frac{1-\pi_{1,0}}{1-\pi_{1,1}} - 1\right). \tag{A.5}$$

By applying the logarithm inequality $1-\frac{1}{x} < \ln(x) < (x-1),\ \forall x > 0$, to (A.5), we have

$$\begin{aligned}
\ln\frac{\pi_{1,1}}{\pi_{1,0}} &> 1-\frac{\pi_{1,0}}{\pi_{1,1}} \\
&= \frac{1-\pi_{1,1}}{\pi_{1,1}}\left(\frac{1-\pi_{1,0}}{1-\pi_{1,1}} - 1\right) \\
&\ge \frac{1-\pi_{1,1}}{\pi_{1,1}} \ln\frac{1-\pi_{1,0}}{1-\pi_{1,1}}.
\end{aligned}$$

Now,

$$\begin{aligned} & \ln\frac{\pi_{1,1}}{\pi_{1,0}} \geq \frac{1-\pi_{1,1}}{\pi_{1,1}}\ln\frac{1-\pi_{1,0}}{1-\pi_{1,1}} \\ \Leftrightarrow \quad & \frac{\ln(\pi_{1,1}/\pi_{1,0})}{\ln((1-\pi_{1,0})/(1-\pi_{1,1}))}\frac{\pi_{1,1}}{1-\pi_{1,1}} \geq 1 \\ \Leftrightarrow \quad & \ln\left(\frac{\ln(\pi_{1,1}/\pi_{1,0})}{\ln((1-\pi_{1,0})/(1-\pi_{1,1}))}\frac{\pi_{1,1}}{1-\pi_{1,1}}\right) \geq 0. \end{aligned}$$

A.2 Proof of Lemma 3.4.1

To show that, for the optimal t^* and $\alpha < 0.5$, Chernoff information, C, is monotonically decreasing function of $P_{1,0}$ while keeping $P_{0,1}$ fixed is equivalent to showing that $\tilde{C}$, is monotonically increasing function of $P_{1,0}$ while keeping $P_{0,1}$ fixed. Differentiating both sides of $\tilde{C}$ with respect to $P_{1,0}$, we get

$$\begin{aligned} \frac{d\tilde{C}}{P_{1,0}} &= \pi_{1,0}^{t^*}\pi_{1,1}^{(1-t^*)}\left(\frac{dt^*}{P_{1,0}}\ln\frac{\pi_{1,0}}{\pi_{1,1}} + (1-t^*)\frac{\pi'_{1,1}}{\pi_{1,1}} + t^*\frac{\pi'_{1,0}}{\pi_{1,0}}\right) \\ &+ (1-\pi_{1,0})^{t^*}(1-\pi_{1,1})^{(1-t^*)}\left(\frac{dt^*}{P_{1,0}}\ln\frac{1-\pi_{1,0}}{1-\pi_{1,1}} - (1-t^*)\frac{\pi'_{1,1}}{1-\pi_{1,1}} - t^*\frac{\pi'_{1,0}}{1-\pi_{1,0}}\right) \end{aligned}$$

In the above equation,

$$\frac{dt^*}{P_{1,0}} = \frac{\left(\ln\frac{\pi_{1,1}}{\pi_{1,0}} + \ln\frac{1-\pi_{1,0}}{1-\pi_{1,1}}\right)\left(\frac{G'}{G} + \frac{\pi'_{1,1}}{\pi_{1,1}} + \frac{\pi'_{1,1}}{1-\pi_{1,1}}\right) - \left(\ln G + \ln\frac{\pi_{1,1}}{1-\pi_{1,1}}\right)\left(\frac{\pi'_{1,1}}{\pi_{1,1}} - \frac{\pi'_{1,0}}{\pi_{1,0}} + \frac{\pi'_{1,1}}{1-\pi_{1,1}} - \frac{\pi'_{1,0}}{1-\pi_{1,0}}\right)}{\left(\ln\frac{\pi_{1,1}}{\pi_{1,0}} + \ln\frac{1-\pi_{1,0}}{1-\pi_{1,1}}\right)^2}$$

where $G = \dfrac{\ln(\pi_{1,1}/\pi_{1,0})}{\ln((1-\pi_{1,0})/(1-\pi_{1,1}))}$ and

$$\frac{G'}{G} = \frac{\ln\frac{1-\pi_{1,0}}{1-\pi_{1,1}}\left(\frac{\pi'_{1,1}}{\pi_{1,1}} - \frac{\pi'_{1,0}}{\pi_{1,0}}\right) - \ln\frac{\pi_{1,1}}{\pi_{1,0}}\left(\frac{\pi'_{1,1}}{1-\pi_{1,1}} - \frac{\pi'_{1,0}}{1-\pi_{1,0}}\right)}{\ln\frac{\pi_{1,1}}{\pi_{1,0}}\ln\frac{1-\pi_{1,0}}{1-\pi_{1,1}}}.$$

Let us denote $a_1 = \ln G + \ln(\pi_{1,1}/(1-\pi_{1,1}))$, $a_2 = \ln(\pi_{1,1}/\pi_{1,0}) + \ln((1-\pi_{1,0})/(1-\pi_{1,1}))$, $b_1 = (\pi'_{1,1}/\pi_{1,1}) + (\pi'_{1,1}/(1-\pi_{1,1}))$, $b_2 = (\pi'_{1,0}/\pi_{1,0}) + (\pi'_{1,0}/(1-\pi_{1,0}))$, $c_1 = \pi_{1,0}^{t^*}\pi_{1,1}^{1-t^*}\ln(\pi_{1,1}/\pi_{1,0})$, $c_2 = (1-\pi_{1,0})^{t^*}(1-\pi_{1,1})^{1-t^*}\ln((1-\pi_{1,0})/(1-\pi_{1,1}))$, $d_1 = ((1-t^*)(\pi'_{1,1}/\pi_{1,1}) + t^*(\pi_{1,0}/\pi_{1,0}))\pi_{1,0}^{t^*}\pi_{1,1}^{1-t^*}$ and $d_2 = ((1-t^*)(\pi'_{1,1}/(1-\pi_{1,1})) + t^*(\pi_{1,0}/(1-\pi_{1,0})))(1-\pi_{1,0})^{t^*}(1-\pi_{1,1})^{1-t^*}$. Now, $\tilde{C}$, is monotonically increasing function of $P_{1,0}$ while keeping $P_{0,1}$ fixed if

$$
\begin{aligned}
& a_1[b_1c_1 + b_2c_2] + a_2[-(G'/G)c_1 + b_1c_2] + a_2^2 d_1 > a_1[b_1c_2 + b_2c_1] + a_2[-(G'/G)c_2 + b_1c_1] + a_2^2 d_2 \\
\Leftrightarrow \quad & a_2^2(d_1 - d_2) > (c_1 - c_2)(a_1(b_2 - b_1) + a_2((G'/G) + b_1)) \\
\Leftrightarrow \quad & a_2^2(d_1 - d_2) > 0
\end{aligned}
$$

where the last inequality follows from the fact that $(c_1 - c_2) = 0$ as given in (3.12).

Now, to show that $\dfrac{d\tilde{C}}{P_{1,0}} > 0$ is equivalent to show that $(d_1 - d_2) > 0$. In other words,

$$
t^*(1-P_f)\left[\left(\frac{\pi_{1,1}}{\pi_{1,0}}\right)^{1-t^*} - \left(\frac{1-\pi_{1,1}}{1-\pi_{1,0}}\right)^{1-t^*}\right] + (1-t^*)(1-P_d)\left[\left(\frac{\pi_{1,0}}{\pi_{1,1}}\right)^{t^*} - \left(\frac{1-\pi_{1,0}}{1-\pi_{1,1}}\right)^{t^*}\right] > 0. \tag{A.6}
$$

Note that,

$$
\left[\left(\frac{\pi_{1,1}}{\pi_{1,0}}\right)^{1-t^*} - \left(\frac{1-\pi_{1,1}}{1-\pi_{1,0}}\right)^{1-t^*}\right] \geq 0; \ \left[\left(\frac{\pi_{1,0}}{\pi_{1,1}}\right)^{t^*} - \left(\frac{1-\pi_{1,0}}{1-\pi_{1,1}}\right)^{t^*}\right] \leq 0.
$$

Hence, (A.6) can be simplified to,

$$
\frac{1-P_f}{1-P_d} > \frac{(1-t^*)\left[\left(\frac{1-\pi_{1,0}}{1-\pi_{1,1}}\right)^{t^*} - \left(\frac{\pi_{1,0}}{\pi_{1,1}}\right)^{t^*}\right]}{t^*\left[\left(\frac{\pi_{1,1}}{\pi_{1,0}}\right)^{1-t^*} - \left(\frac{1-\pi_{1,1}}{1-\pi_{1,0}}\right)^{1-t^*}\right]}. \tag{A.7}
$$

Similarly, for the optimal t^* and $\alpha < 0.5$, Chernoff information, C, is monotonically decreasing

function of $P_{0,1}$ while keeping $P_{1,0}$ fixed if $(d_1 - d_2) > 0$, which is equivalent to show that,

$$t^*(-P_f)\left[\left(\frac{\pi_{1,1}}{\pi_{1,0}}\right)^{1-t^*} - \left(\frac{1-\pi_{1,1}}{1-\pi_{1,0}}\right)^{1-t^*}\right] + (1-t^*)(-P_d)\left[\left(\frac{\pi_{1,0}}{\pi_{1,1}}\right)^{t^*} - \left(\frac{1-\pi_{1,0}}{1-\pi_{1,1}}\right)^{t^*}\right] > 0. \tag{A.8}$$

Furthermore, (A.8) can be simplified to

$$\frac{P_f}{P_d} < \frac{(1-t^*)\left[\left(\frac{1-\pi_{1,0}}{1-\pi_{1,1}}\right)^{t^*} - \left(\frac{\pi_{1,0}}{\pi_{1,1}}\right)^{t^*}\right]}{t^*\left[\left(\frac{\pi_{1,1}}{\pi_{1,0}}\right)^{1-t^*} - \left(\frac{1-\pi_{1,1}}{1-\pi_{1,0}}\right)^{1-t^*}\right]}. \tag{A.9}$$

Combining (A.7) and (A.9), the condition to make Lemma 3.4.1 true becomes

$$\frac{P_f}{P_d} < \frac{(1-t^*)\left[\left(\frac{1-\pi_{1,0}}{1-\pi_{1,1}}\right)^{t^*} - \left(\frac{\pi_{1,0}}{\pi_{1,1}}\right)^{t^*}\right]}{t^*\left[\left(\frac{\pi_{1,1}}{\pi_{1,0}}\right)^{1-t^*} - \left(\frac{1-\pi_{1,1}}{1-\pi_{1,0}}\right)^{1-t^*}\right]} < \frac{1-P_f}{1-P_d}. \tag{A.10}$$

Note that right hand inequality in (A.10) can be rewritten as

$$\begin{aligned}
&\left(\frac{1}{t^*}-1\right)\left[\left(\frac{1-\pi_{1,0}}{1-\pi_{1,1}}\right)^{t^*} - \left(\frac{\pi_{1,0}}{\pi_{1,1}}\right)^{t^*}\right] < \frac{1-P_f}{1-P_d}\left[\left(\frac{\pi_{1,1}}{\pi_{1,0}}\right)^{1-t^*} - \left(\frac{1-\pi_{1,1}}{1-\pi_{1,0}}\right)^{1-t^*}\right] \\
\Leftrightarrow\ &\left(\frac{1}{t^*}-1\right)\left[\left(\frac{1-\pi_{1,0}}{1-\pi_{1,1}}\right)^{t^*} - \left(\frac{\pi_{1,0}}{\pi_{1,1}}\right)^{t^*}\right] < \frac{1-P_f}{1-P_d}\left[\left(\frac{\pi_{1,1}}{\pi_{1,0}}\right)\left(\frac{\pi_{1,0}}{\pi_{1,1}}\right)^{t^*} - \left(\frac{1-\pi_{1,1}}{1-\pi_{1,0}}\right)\left(\frac{1-\pi_{1,0}}{1-\pi_{1,1}}\right)^{t^*}\right] \\
\Leftrightarrow\ &\left(\frac{1-\pi_{1,0}}{1-\pi_{1,1}}\right)^{t^*}\left[\frac{1-P_f}{1-P_d}\left(\frac{1-\pi_{1,1}}{1-\pi_{1,0}}\right) + \left(\frac{1}{t^*}-1\right)\right] < \left(\frac{\pi_{1,0}}{\pi_{1,1}}\right)^{t^*}\left[\frac{1-P_f}{1-P_d}\left(\frac{\pi_{1,1}}{\pi_{1,0}}\right) + \left(\frac{1}{t^*}-1\right)\right] \\
\Leftrightarrow\ &\left(\frac{(1/\pi_{1,0})-1}{(1/\pi_{1,1})-1}\right)^{t^*}\left[\frac{1-P_f}{1-P_d}\left(\frac{1-\pi_{1,1}}{1-\pi_{1,0}}\right) + \left(\frac{1}{t^*}-1\right)\right] < \left[\frac{1-P_f}{1-P_d}\left(\frac{\pi_{1,1}}{\pi_{1,0}}\right) + \left(\frac{1}{t^*}-1\right)\right].
\end{aligned}$$

Using the result from (3.12), the above equation can be written as

$$\frac{\ln(\pi_{1,1}/\pi_{1,0})}{\ln\left(\frac{(1-\pi_{1,0})}{(1-\pi_{1,1})}\right)}\left(\frac{\pi_{1,1}}{1-\pi_{1,1}}\right) < \frac{\left[\frac{1-P_f}{1-P_d}\left(\frac{\pi_{1,1}}{\pi_{1,0}}\right) + \left(\frac{1}{t^*}-1\right)\right]}{\left[\frac{1-P_f}{1-P_d}\left(\frac{1-\pi_{1,1}}{1-\pi_{1,0}}\right) + \left(\frac{1}{t^*}-1\right)\right]}.$$

Using the fact that $G = \dfrac{\ln(\pi_{1,1}/\pi_{1,0})}{\ln\left(\dfrac{(1-\pi_{1,0})}{(1-\pi_{1,1})}\right)}$, we get

$$G < \frac{\left[\dfrac{1-P_f}{1-P_d}\left(\dfrac{1}{\pi_{1,0}}\right) + \left(\dfrac{1}{t^*}-1\right)\left(\dfrac{1}{\pi_{1,1}}\right)\right]}{\left[\dfrac{1-P_f}{1-P_d}\left(\dfrac{1}{1-\pi_{1,0}}\right) + \left(\dfrac{1}{t^*}-1\right)\left(\dfrac{1}{1-\pi_{1,1}}\right)\right]}.$$

After some simplification the above condition can be written as

$$\begin{aligned} &\frac{1-P_f}{1-P_d}\left[\frac{G}{1-\pi_{1,0}} - \frac{1}{\pi_{1,0}}\right] < \left(\frac{1}{t^*}-1\right)\left[\frac{1}{\pi_{1,1}} - \frac{G}{1-\pi_{1,1}}\right] \\ \Leftrightarrow \quad &\left(\frac{1-P_f}{1-P_d}\right)\left(\frac{\pi_{1,1}}{\pi_{1,0}}\right)\left(\frac{1-\pi_{1,1}}{1-\pi_{1,0}}\right)[\pi_{1,0}(G+1)-1] < \left(\frac{1}{t^*}-1\right)[1-\pi_{1,1}(G+1)] \end{aligned}$$

$$\frac{1}{t^*}[\pi_{1,1}(G+1)-1] < \left(\frac{1-P_f}{1-P_d}\right)\left(\frac{\pi_{1,1}}{\pi_{1,0}}\right)\left(\frac{1-\pi_{1,1}}{1-\pi_{1,0}}\right)[1-\pi_{1,0}(G+1)] + [\pi_{1,1}(G+1)-1]. \quad \text{(A.11)}$$

Notice that, in the above equation

$$\pi_{1,1}(G+1) \geq 1 \text{ and } \pi_{1,0}(G+1) \leq 1 \quad \text{(A.12)}$$

or equivalently $\dfrac{1-\pi_{1,1}}{\pi_{1,1}} \leq G \leq \dfrac{1-\pi_{1,0}}{\pi_{1,0}}$. The second inequality in (A.12) follows from the fact that $\ln\left(\dfrac{\pi_{1,1}}{\pi_{1,0}}\right) \geq \ln\left(\frac{1-\pi_{1,0}}{1-\pi_{1,1}}\right)\dfrac{1-\pi_{1,1}}{\pi_{1,1}}$. Using logarithm inequality, we have $\ln\left(\dfrac{\pi_{1,1}}{\pi_{1,0}}\right) \geq \left(1-\dfrac{\pi_{1,0}}{\pi_{1,1}}\right) = \left(\dfrac{1-\pi_{1,1}}{\pi_{1,1}}\right)\left(\dfrac{1-\pi_{1,0}}{1-\pi_{1,1}}-1\right) \geq \ln\left(\dfrac{1-\pi_{1,0}}{1-\pi_{1,1}}\right)\dfrac{1-\pi_{1,1}}{\pi_{1,1}}$. Similarly, to show that the second inequality in (A.12) is true we show $\ln\left(\dfrac{\pi_{1,1}}{\pi_{1,0}}\right) \leq \ln\left(\dfrac{1-\pi_{1,0}}{1-\pi_{1,1}}\right)\dfrac{1-\pi_{1,0}}{\pi_{1,0}}$. Using logarithm inequality, $\ln\left(\dfrac{\pi_{1,1}}{\pi_{1,0}}\right) \leq \left(\dfrac{\pi_{1,1}}{\pi_{1,0}}-1\right) = \left(\dfrac{1-\pi_{1,0}}{\pi_{1,0}}\right)\left(1-\dfrac{1-\pi_{1,1}}{1-\pi_{1,0}}\right) \leq \ln\left(\dfrac{1-\pi_{1,0}}{1-\pi_{1,1}}\right)\dfrac{1-\pi_{1,0}}{\pi_{1,0}}$. Using these results we can then write (A.11) in the form below,

$$\frac{[\pi_{1,1}(G+1)-1]}{\left(\dfrac{1-P_f}{1-P_d}\right)\left(\dfrac{\pi_{1,1}}{\pi_{1,0}}\right)\left(\dfrac{1-\pi_{1,1}}{1-\pi_{1,0}}\right)[1-\pi_{1,0}(G+1)] + [\pi_{1,1}(G+1)-1]} < t^*$$

$$\Leftrightarrow \frac{1}{\left(\frac{1-P_f}{1-P_d}\right)\left(\frac{\pi_{1,1}}{\pi_{1,0}}\right)\left(\frac{1-\pi_{1,1}}{1-\pi_{1,0}}\right)\frac{[1-\pi_{1,0}(G+1)]}{[\pi_{1,1}(G+1)-1]}+1} < t^*. \tag{A.13}$$

Similarly, the left hand side inequality in (A.10) can be written as,

$$\begin{aligned}
&\frac{P_f}{P_d}\left[\left(\frac{\pi_{1,1}}{\pi_{1,0}}\right)^{1-t^*}-\left(\frac{1-\pi_{1,1}}{1-\pi_{1,0}}\right)^{1-t^*}\right] < \left(\frac{1}{t^*}-1\right)\left[\left(\frac{1-\pi_{1,0}}{1-\pi_{1,1}}\right)^{t^*}-\left(\frac{\pi_{1,0}}{\pi_{1,1}}\right)^{t^*}\right] \\
\Leftrightarrow\ &\frac{P_f}{P_d}\left[\left(\frac{\pi_{1,1}}{\pi_{1,0}}\right)\left(\frac{\pi_{1,0}}{\pi_{1,1}}\right)^{t^*}-\left(\frac{1-\pi_{1,1}}{1-\pi_{1,0}}\right)\left(\frac{1-\pi_{1,0}}{1-\pi_{1,1}}\right)^{t}\right] < \left(\frac{1}{t^*}-1\right)\left[\left(\frac{1-\pi_{1,0}}{1-\pi_{1,1}}\right)^{t^*}-\left(\frac{\pi_{1,0}}{\pi_{1,1}}\right)^{t^*}\right] \\
\Leftrightarrow\ &\left(\frac{\pi_{1,0}}{\pi_{1,1}}\right)^{t}\left[\frac{P_f}{P_d}\left(\frac{\pi_{1,1}}{\pi_{1,0}}\right)+\left(\frac{1}{t^*}-1\right)\right] < \left(\frac{1-\pi_{1,0}}{1-\pi_{1,1}}\right)^{t}\left[\frac{P_f}{P_d}\left(\frac{1-\pi_{1,1}}{1-\pi_{1,0}}\right)+\left(\frac{1}{t^*}-1\right)\right] \\
\Leftrightarrow\ &\left[\frac{P_f}{P_d}\left(\frac{\pi_{1,1}}{\pi_{1,0}}\right)+\left(\frac{1}{t^*}-1\right)\right] < \left(\frac{(1/\pi_{1,0})-1}{(1/\pi_{1,1})-1}\right)^{t^*}\left[\frac{P_f}{P_d}\left(\frac{1-\pi_{1,1}}{1-\pi_{1,0}}\right)+\left(\frac{1}{t^*}-1\right)\right].
\end{aligned}$$

Using the results from (3.12), the above equation can be written as,

$$\frac{\left[\frac{P_f}{P_d}\left(\frac{\pi_{1,1}}{\pi_{1,0}}\right)+\left(\frac{1}{t^*}-1\right)\right]}{\left[\frac{P_f}{P_d}\left(\frac{1-\pi_{1,1}}{1-\pi_{1,0}}\right)+\left(\frac{1}{t^*}-1\right)\right]} < \frac{\ln(\pi_{1,1}/\pi_{1,0})}{\ln\left(\frac{(1-\pi_{1,0})}{(1-\pi_{1,1})}\right)}\left(\frac{\pi_{1,1}}{1-\pi_{1,1}}\right).$$

Lets denote $G = \dfrac{\ln(\pi_{1,1}/\pi_{1,0})}{\ln\left(\frac{(1-\pi_{1,0})}{(1-\pi_{1,1})}\right)}$, we get

$$\frac{\left[\frac{P_f}{P_d}\left(\frac{1}{\pi_{1,0}}\right)+\left(\frac{1}{t^*}-1\right)\left(\frac{1}{\pi_{1,1}}\right)\right]}{\left[\frac{P_f}{P_d}\left(\frac{1}{1-\pi_{1,0}}\right)+\left(\frac{1}{t^*}-1\right)\left(\frac{1}{1-\pi_{1,1}}\right)\right]} < G.$$

After some simplification the above condition can be written as,

$$\begin{aligned}
&\left(\frac{1}{t^*}-1\right)\left[\frac{1}{\pi_{1,1}}-\frac{G}{1-\pi_{1,1}}\right] < \frac{P_f}{P_d}\left[\frac{G}{1-\pi_{1,0}}-\frac{1}{\pi_{1,0}}\right] \\
\Leftrightarrow\ &\left(\frac{1}{t^*}-1\right)[1-\pi_{1,1}(G+1)] < \left(\frac{P_f}{P_d}\right)\left(\frac{\pi_{1,1}}{\pi_{1,0}}\right)\left(\frac{1-\pi_{1,1}}{1-\pi_{1,0}}\right)[\pi_{1,0}(G+1)-1] \\
\Leftrightarrow\ &\left(\frac{P_f}{P_d}\right)\left(\frac{\pi_{1,1}}{\pi_{1,0}}\right)\left(\frac{1-\pi_{1,1}}{1-\pi_{1,0}}\right)[1-\pi_{1,0}(G+1)]+[\pi_{1,1}(G+1)-1] < \frac{1}{t^*}[\pi_{1,1}(G+1)-1].
\end{aligned}$$

Using (A.12), the condition can be written as

$$t^* < \frac{[\pi_{1,1}(G+1)-1]}{\left(\frac{P_f}{P_d}\right)\left(\frac{\pi_{1,1}}{\pi_{1,0}}\right)\left(\frac{1-\pi_{1,1}}{1-\pi_{1,0}}\right)[1-\pi_{1,0}(G+1)]+[\pi_{1,1}(G+1)-1]}$$

$$t^* < \frac{1}{\left(\frac{P_f}{P_d}\right)\left(\frac{\pi_{1,1}}{\pi_{1,0}}\right)\left(\frac{1-\pi_{1,1}}{1-\pi_{1,0}}\right)\frac{[1-\pi_{1,0}(G+1)]}{[\pi_{1,1}(G+1)-1]}+1}. \tag{A.14}$$

Now from (A.13) and (A.14), Lemma 3.4.1 is true if

$$A = \frac{1}{\left(\frac{1-P_f}{1-P_d}\right)\left(\frac{\pi_{1,1}}{\pi_{1,0}}\right)\left(\frac{1-\pi_{1,1}}{1-\pi_{1,0}}\right)\frac{[1-\pi_{1,0}(G+1)]}{[\pi_{1,1}(G+1)-1]}+1} < t^* < \frac{1}{\frac{P_f}{P_d}\left(\frac{\pi_{1,1}}{\pi_{1,0}}\right)\left(\frac{1-\pi_{1,1}}{1-\pi_{1,0}}\right)\frac{[1-\pi_{1,0}(G+1)]}{[\pi_{1,1}(G+1)-1]}+1} = B. \tag{A.15}$$

Next we show that, the optimal t^* is with in the region (A, B). Using the results from (A.28), we start from the inequality

$$\begin{aligned} & \frac{P_f}{P_d}\frac{\pi_{1,1}}{\pi_{1,0}} < 1 < \frac{1-P_f}{1-P_d}\frac{1-\pi_{1,1}}{1-\pi_{1,0}} \\ \Leftrightarrow \quad & \frac{P_f}{P_d}\frac{\pi_{1,1}}{\pi_{1,0}}\frac{[1-\pi_{1,0}(G+1)]}{[\pi_{1,1}(G+1)-1]} < \frac{1-\pi_{1,0}(G+1)}{\pi_{1,1}(G+1)-1} < \frac{1-P_f}{1-P_d}\frac{1-\pi_{1,1}}{1-\pi_{1,0}}\frac{[1-\pi_{1,0}(G+1)]}{[\pi_{1,1}(G+1)-1]} \end{aligned}$$

Let us denote, $Y = \left(\frac{\pi_{1,1}}{\pi_{1,0}}\right)\left(\frac{1-\pi_{1,1}}{1-\pi_{1,0}}\right)\frac{[1-\pi_{1,0}(G+1)]}{[\pi_{1,1}(G+1)-1]}$, then the above condition can be written as,

$$\frac{P_f}{P_d}Y\frac{1-\pi_{1,0}}{1-\pi_{1,1}} < \frac{1-\pi_{1,0}(G+1)}{\pi_{1,1}(G+1)-1} < \frac{1-P_f}{1-P_d}Y\frac{\pi_{1,0}}{\pi_{1,1}} \tag{A.16}$$

Next, we use the log inequality, $\frac{x-1}{x} < \ln(x) < (x-1),\ \forall x > 0$, to derive further results. Let

us focus our attention to the left hand side inequality in (A.16)

$$\begin{aligned} & \frac{P_f}{P_d} Y \frac{1-\pi_{1,0}}{1-\pi_{1,1}} < \frac{1-\pi_{1,0}(G+1)}{\pi_{1,1}(G+1)-1} \\ \Leftrightarrow \quad & P_f Y \left[\frac{G\pi_{1,1}}{1-\pi_{1,1}} - 1\right] < P_d \left[1 - \frac{G\pi_{1,0}}{1-\pi_{1,0}}\right] \\ \Leftrightarrow \quad & P_f Y \ln\left(G\frac{\pi_{1,1}}{1-\pi_{1,1}}\right) < P_d \ln\left(\frac{1}{G}\frac{1-\pi_{1,0}}{\pi_{1,0}}\right) \end{aligned} \tag{A.17}$$

Now, let us focus our attention to the right hand side inequality in (A.16)

$$\begin{aligned} & \frac{1-\pi_{1,0}(G+1)}{\pi_{1,1}(G+1)-1} < \frac{1-P_f}{1-P_d} Y \frac{\pi_{1,0}}{\pi_{1,1}} \\ \Leftrightarrow \quad & (1-P_d)\left(\frac{1-\pi_{1,0}}{G\pi_{1,0}} - 1\right) < (1-P_f) Y \left(1 - \frac{1-\pi_{1,1}}{G\pi_{1,1}}\right) \\ \Leftrightarrow \quad & (1-P_d) \ln\left(\frac{1}{G}\frac{1-\pi_{1,0}}{\pi_{1,0}}\right) < (1-P_f) Y \ln\left(G\frac{\pi_{1,1}}{1-\pi_{1,1}}\right) \end{aligned} \tag{A.18}$$

Now using the results from (A.17) and (A.18), we can deduce that

$$\begin{aligned} & \left(\frac{P_f}{P_d}\right) Y < \frac{\ln\left(\frac{1}{G}\frac{1-\pi_{1,0}}{\pi_{1,0}}\right)}{\ln\left(G\frac{\pi_{1,1}}{1-\pi_{1,1}}\right)} < \left(\frac{1-P_f}{1-P_d}\right) Y \\ \Leftrightarrow \quad & \frac{1}{1+\left(\frac{1-P_f}{1-P_d}\right) Y} < \frac{1}{1+\frac{\ln\left(\frac{1}{G}\frac{1-\pi_{1,0}}{\pi_{1,0}}\right)}{\ln\left(G\frac{\pi_{1,1}}{1-\pi_{1,1}}\right)}} < \frac{1}{1+\left(\frac{P_f}{P_d}\right) Y} \end{aligned} \tag{A.19}$$

which is true from the fact that for $a > 0, b > 0$, $\frac{1}{1+a} < \frac{1}{1+b}$ iff $b < a$. Next, observe that, t^* as given in (3.13) can be written as

$$t^* = \frac{\ln\left(G\frac{\pi_{1,1}}{1-\pi_{1,1}}\right)}{\ln\left(\frac{(1/\pi_{1,0})-1}{(1/\pi_{1,1})-1}\right)} = \frac{\ln(G) + \ln\left(\frac{\pi_{1,1}}{1-\pi_{1,1}}\right)}{\ln\left(\frac{\pi_{1,1}}{\pi_{1,0}}\right) + \ln\left(\frac{1-\pi_{1,0}}{1-\pi_{1,1}}\right)}.$$

Observe that, $\ln\left(G\frac{\pi_{1,1}}{1-\pi_{1,1}}\right) \geq 0$ and $\ln\left(\frac{1}{G}\frac{1-\pi_{1,0}}{\pi_{1,0}}\right) \geq 0$ or equivalently $\left(G\frac{\pi_{1,1}}{1-\pi_{1,1}}\right) \geq 1$ and $\left(\frac{1}{G}\frac{1-\pi_{1,0}}{\pi_{1,0}}\right) \geq 1$ from (A.12). Now,

$$t^* = \frac{1}{1+\frac{\ln\left(\frac{\pi_{1,1}}{\pi_{1,0}}\right)+\ln\left(\frac{1-\pi_{1,0}}{1-\pi_{1,1}}\right)-\ln(G)-\ln\left(\frac{\pi_{1,1}}{1-\pi_{1,1}}\right)}{\ln(G)+\ln\left(\frac{\pi_{1,1}}{1-\pi_{1,1}}\right)}} = \frac{1}{1+\frac{\ln\left(\frac{1}{G}\frac{1-\pi_{1,0}}{\pi_{1,0}}\right)}{\ln\left(G\frac{\pi_{1,1}}{1-\pi_{1,1}}\right)}}.$$

Which along with (A.19) implies that

$$\frac{1}{1+\left(\frac{1-P_f}{1-P_d}\right)Y} < t^* < \frac{1}{1+\left(\frac{P_f}{P_d}\right)Y}$$

or in other words, $A < t^* < B$. This completes our proof.

A.3 Sensitivity to Imperfect Knowledge

In this section, we discuss the sensitivity of system performance to imperfect knowledge regarding the fraction of Byzantines α in the network and the prior probability of hypotheses, i.e., $(P_0,\ P_1)$. We limit the analysis to a couple of illustrative examples.

In many practical scenarios, the value of the fraction of Byzantines α in the network might not be known a-priori. In such scenarios, α may be estimated (learned) by observing decisions at the FC over a fixed duration. Next, we present a rather simple estimation procedure and some numerical results to corroborate our claim.

We assume that $P_d = 0.8$, $P_f = 0.2$ and the fraction of Byzantines is $\alpha = 0.2$ with $(P_{1,0}, P_{0,1}) = (1, 1)$. Based on the received decisions under hypothesis H_1, the FC can estimate $\hat{\alpha}$ as follows:

$$\hat{\alpha} = \frac{P_d - \pi_{1,1}}{2P_d - 1},$$

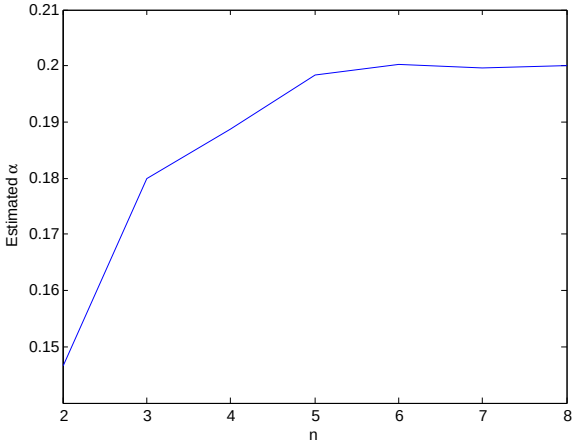


Fig. A.1: Estimation of the fraction of Byzantines as a function of $N = 10^n$ when the true value of $\alpha = 0.2$.

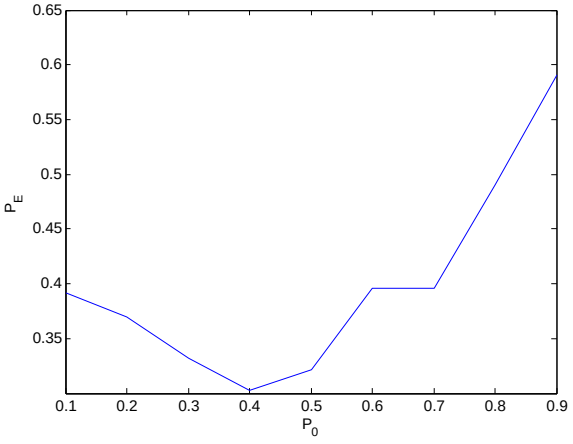


Fig. A.2: Error probability in the presence of imperfect knowledge of P_0.

where $\pi_{1,1}$ is the fraction of 1's received at the FC. In Figure A.1, we plot the value of estimated α at the FC as a function of the number of decisions at the FC, i.e., $N = 10^n$.

It can be seen from Figure A.1 that the estimated α approaches the true value of α as the number of decisions N at the FC increases.

Next, we look at the sensitivity of the performance of the detection scheme to the uncertainty regarding the prior probability of hypotheses, i.e., $(P_0,\ P_1)$. Sensitivity of the performance of the scheme to the uncertainty of parameter values is a model mismatch problem. In general, finding the analytical expressions for performance degradation due to model mismatch is a difficult problem, thus, we limit our analysis to numerical results. However, one can expect that the performance of the scheme will improve as the estimated parameter values approach their true value.

In Figure A.2, we plot the probability of error as the value of P_0 at the FC is varied from 0.1 to 0.9 when the actual value of P_0 is 0.4, $N = 10$ and $(P_d, P_f) = (0.8, 0.1)$ with $(P_{1,0}, P_{0,1}) = (1, 1)$. Note that, the error probability is minimum when the estimated P_0 is equal to the actual P_0.

A.4 Proof of $\dfrac{dr\left(P_{1,0}, K^*, \alpha\right)}{dP_{1,0}} > 0$

Differentiating both sides of $r\left(P_{1,0}, K^*, \alpha\right)$ with respect to $P_{1,0}$, we get

$$\frac{dr\left(P_{1,0}, K^*, \alpha\right)}{dP_{1,0}} = (K^* - 1)\alpha\left(\frac{1-P_f}{\pi_{1,0}} - \frac{1-P_d}{\pi_{1,1}}\right) - (N - K^*)\alpha\left(\frac{1-P_f}{1-\pi_{1,0}} - \frac{1-P_d}{1-\pi_{1,1}}\right).$$

In the following we show that

$$\frac{dr\left(P_{1,0}, K^*, \alpha\right)}{dP_{1,0}} > 0 \tag{A.20}$$

i.e., $r\left(P_{1,0}, K^*, \alpha\right)$ is non-decreasing. Observe that in the above equation,

$$\frac{(1-P_f)}{\pi_{1,0}} > \frac{(1-P_d)}{\pi_{1,1}}. \tag{A.21}$$

To show that the above condition is true, we start from the inequality

$$P_d > P_f \tag{A.22}$$

$$\Leftrightarrow \quad \frac{P_d}{1-P_d} > \frac{P_f}{1-P_f} \tag{A.23}$$

$$\Leftrightarrow \quad \alpha P_{1,0} + (1 - P_{0,1}\alpha)\frac{P_d}{1-P_d} > \alpha P_{1,0} + (1 - P_{0,1}\alpha)\frac{P_f}{1-P_f} \tag{A.24}$$

$$\Leftrightarrow \quad \frac{\alpha P_{1,0}(1-P_d) + P_d(1-P_{0,1}\alpha)}{(1-P_d)} > \frac{\alpha P_{1,0}(1-P_f) + P_f(1-P_{0,1}\alpha)}{(1-P_f)} \tag{A.25}$$

$$\Leftrightarrow \quad \frac{\pi_{1,1}}{(1-P_d)} > \frac{\pi_{1,0}}{(1-P_f)} \tag{A.26}$$

$$\Leftrightarrow \quad \frac{(1-P_f)}{\pi_{1,0}} > \frac{(1-P_d)}{\pi_{1,1}} \tag{A.27}$$

Similarly, it can be shown that

$$\frac{1-\pi_{1,1}}{1-P_d} > \frac{1-\pi_{1,0}}{1-P_f} \tag{A.28}$$

Now from (A.21) and (A.28), to show that $\frac{dr\left(P_{1,0}, K^*, \alpha\right)}{dP_{1,0}} > 0$ is equivalent to show that

$$(K^* - 1)\left(\frac{1-P_f}{\pi_{1,0}} - \frac{1-P_d}{\pi_{1,1}}\right) > (N - K^*)\left(\frac{1-P_f}{1-\pi_{1,0}} - \frac{1-P_d}{1-\pi_{1,1}}\right) \tag{A.29}$$

Next, we consider two different cases, first when there are odd number of nodes in the network and second when there are even number of nodes in the network.

Odd Number of Nodes: When there are odd number of nodes in the network, the majority fusion rule is $K^* = (N+1)/2$. In this case (A.29) is equivalent to show that

$$\left(\frac{N-1}{2}\right)\left(\frac{1-P_f}{\pi_{1,0}} - \frac{1-P_d}{\pi_{1,1}}\right) > \left(\frac{N-1}{2}\right)\left(\frac{1-P_f}{1-\pi_{1,0}} - \frac{1-P_d}{1-\pi_{1,1}}\right). \tag{A.30}$$

To show that the above condition is true, we start from the following inequality

$$\begin{aligned} & \frac{(1-\pi_{1,0})(1-\pi_{1,1})}{\pi_{1,0}\pi_{1,1}} > -1 \\ \Leftrightarrow \quad & \left[\frac{1}{\pi_{1,0}} - \frac{1}{\pi_{1,1}}\right] > \left[\frac{1}{1-\pi_{1,0}} - \frac{1}{1-\pi_{1,1}}\right] \\ \Leftrightarrow \quad & \left[\frac{1}{\pi_{1,0}} - \frac{1}{1-\pi_{1,0}}\right] > \left[\frac{1}{\pi_{1,1}} - \frac{1}{1-\pi_{1,1}}\right] \end{aligned}$$

Since $\frac{1-P_f}{1-P_d} > 1$, $\pi_{1,0} < 0.5$ (consequence of our assumption) and $N \geq 2$, the above condition is equivalent to

$$\begin{aligned} & \frac{1-P_f}{1-P_d}\left[\frac{1}{\pi_{1,0}} - \frac{1}{1-\pi_{1,0}}\right] > \left[\frac{1}{\pi_{1,1}} - \frac{1}{1-\pi_{1,1}}\right] \\ \Leftrightarrow \quad & \left(\frac{1-P_f}{\pi_{1,0}} - \frac{1-P_d}{\pi_{1,1}}\right) > \left(\frac{1-P_f}{1-\pi_{1,0}} - \frac{1-P_d}{1-\pi_{1,1}}\right) \\ \Leftrightarrow \quad & \left(\frac{N-1}{2}\right)\left(\frac{1-P_f}{\pi_{1,0}} - \frac{1-P_d}{\pi_{1,1}}\right) > \left(\frac{N-1}{2}\right)\left(\frac{1-P_f}{1-\pi_{1,0}} - \frac{1-P_d}{1-\pi_{1,1}}\right) \end{aligned} \tag{A.31}$$

which implies that $\frac{dr\left(P_{1,0}, K^*, \alpha\right)}{dP_{1,0}} > 0$ for odd number of nodes case. Next, we consider the even number of nodes case.

Even Number of Nodes: Now, we consider the case when there are even number of nodes in the network and majority fusion rule is given by $K^* = \frac{N}{2} + 1$. Condition (A.29) is equivalent to show that

$$\left(\frac{N}{2}\right)\left(\frac{1-P_f}{\pi_{1,0}} - \frac{1-P_d}{\pi_{1,1}}\right) > \left(\frac{N}{2} - 1\right)\left(\frac{1-P_f}{1-\pi_{1,0}} - \frac{1-P_d}{1-\pi_{1,1}}\right).$$

Which follows from the fact that

$$\left(\frac{N}{2}\right)\left(\frac{1-P_f}{\pi_{1,0}} - \frac{1-P_d}{\pi_{1,1}}\right) > \left(\frac{N}{2} - 1\right)\left(\frac{1-P_f}{\pi_{1,0}} - \frac{1-P_d}{\pi_{1,1}}\right)$$

and the result given in (A.30). This completes our proof.

A.5 Calculating partial derivative of P_E w.r.t. $P_{1,0}$

First, we calculate the partial derivative of Q_F with respect to $P_{1,0}$. Notice that,

$$Q_F = \sum_{i=K^*}^{N} \left(\begin{array}{c} N \\ i \end{array} \right) (\pi_{1,0})^i (1-\pi_{1,0})^{N-i} \tag{A.32}$$

$$\pi_{1,0} = \alpha(P_{1,0}(1-P_f) + (1-P_{0,1})P_f) + (1-\alpha)P_f \tag{A.33}$$

$$(\pi_{1,0})' = d\pi_{1,0}/dP_{1,0} = \alpha(1-P_f). \tag{A.34}$$

Differentiating both sides of (A.32) with respect to $P_{1,0}$, we get

$$\begin{aligned}
\frac{dQ_F}{dP_{1,0}} &= \left(\begin{array}{c} N \\ K^* \end{array} \right) (K^*(\pi_{1,0})^{K^*-1}(\pi_{1,0})'(1-\pi_{1,0})^{N-K^*} - (\pi_{1,0})^{K^*}(N-K^*)(1-\pi_{1,0})^{N-K^*-1}(\pi_{1,0})') \\
&+ \left(\begin{array}{c} N \\ K^*+1 \end{array} \right) ((K^*+1)(\pi_{1,0})^{K^*}(\pi_{1,0})'(1-\pi_{1,0})^{N-K^*-1} - (\pi_{1,0})^{K^*+1}(N-K^*-1) \\
&(\quad 1-\pi_{1,0})^{N-K^*-2}(\pi_{1,0})') + \cdots + \left(\begin{array}{c} N \\ N \end{array} \right) (N(\pi_{1,0})^{N-1}(\pi_{1,0})' - 0) \\
&= (\pi_{1,0})'(\pi_{1,0})^{K^*-1}(1-\pi_{1,0})^{N-K^*} \left[\left(\begin{array}{c} N \\ K^* \end{array} \right) \left(K^* - \frac{\pi_{1,0}}{1-\pi_{1,0}}(N-K^*) \right) \right. \\
&+ \left. \left(\begin{array}{c} N \\ K^*+1 \end{array} \right) \left((K^*+1)\frac{\pi_{1,0}}{1-\pi_{1,0}} - (N-K^*-1)\left(\frac{\pi_{1,0}}{1-\pi_{1,0}} \right)^2 \right) + \cdots \right] \\
&= (\pi_{1,0})'(\pi_{1,0})^{K^*-1}(1-\pi_{1,0})^{N-K^*} \left[\left(\begin{array}{c} N \\ K^* \end{array} \right) (K^* - \frac{\pi_{1,0}}{1-\pi_{1,0}}(N-K^*)) \right. \\
&+ \left. \frac{\pi_{1,0}}{1-\pi_{1,0}} \left(\begin{array}{c} N \\ K^*+1 \end{array} \right) \left((K^*+1) - (N-K^*-1)\frac{\pi_{1,0}}{1-\pi_{1,0}} \right) + \cdots \right] \\
&= (\pi_{1,0})'(\pi_{1,0})^{K^*-1}(1-\pi_{1,0})^{N-K^*} \left[\left(\begin{array}{c} N \\ K^* \end{array} \right) K^* + \left[-\frac{\pi_{1,0}}{1-\pi_{1,0}} \left(\begin{array}{c} N \\ K^* \end{array} \right) (N-K^*) \right. \right. \\
&+ \left. \left. \frac{\pi_{1,0}}{1-\pi_{1,0}} \left(\begin{array}{c} N \\ K^*+1 \end{array} \right) (K^*+1) \right] + \cdots \right]
\end{aligned}$$

Since, $\begin{pmatrix} N \\ K^* \end{pmatrix} \frac{K^*}{N} = \begin{pmatrix} N-1 \\ K^*-1 \end{pmatrix}$, the above equation can be written as

$$\begin{aligned} \frac{dQ_F}{dP_{1,0}} &= (\pi_{1,0})'(\pi_{1,0})^{K^*-1}(1-\pi_{1,0})^{N-K^*}\Bigg[\begin{pmatrix} N-1 \\ K^*-1 \end{pmatrix} N \\ &+ \frac{\pi_{1,0}}{1-\pi_{1,0}}\Bigg\{\begin{pmatrix} N \\ K^*+1 \end{pmatrix}(K^*+1) - \begin{pmatrix} N \\ K^* \end{pmatrix}(N-K^*)\Bigg\} + \cdots\Bigg]. \quad \text{(A.35)} \end{aligned}$$

Notice that, for any positive integer t

$$\left(\frac{\pi_{1,0}}{1-\pi_{1,0}}\right)^t \left[\begin{pmatrix} N \\ K^*+t \end{pmatrix}(K^*+t) - \begin{pmatrix} N \\ K^*+t-1 \end{pmatrix}(N-K^*-t+1)\right] = 0. \quad \text{(A.36)}$$

Using the result from (A.36), (A.35) can be written as

$$\begin{aligned} &\frac{dQ_F}{dP_{1,0}} = (\pi_{1,0})'(\pi_{1,0})^{K^*-1}(1-\pi_{1,0})^{N-K^*}\left[\begin{pmatrix} N-1 \\ K^*-1 \end{pmatrix} N + \frac{\pi_{1,0}}{1-\pi_{1,0}}[0] + \cdots + [0]\right] \\ \Leftrightarrow\ &\frac{dQ_F}{dP_{1,0}} = \alpha(1-P_f)N\begin{pmatrix} N-1 \\ K^*-1 \end{pmatrix}(\pi_{1,0})^{K^*-1}(1-\pi_{1,0})^{N-K^*}. \end{aligned}$$

Similarly, the partial derivative of Q_D w.r.t. $P_{1,0}$ can calculated to be

$$\frac{dQ_D}{dP_{1,0}} = \alpha(1-P_d)N\begin{pmatrix} N-1 \\ K^*-1 \end{pmatrix}(\pi_{1,1})^{K^*-1}(1-\pi_{1,1})^{N-K^*}.$$

A.6

We want to show that the set $\{B_k\}_{k=1}^K$ can blind the FC if any of following two cases is true.

1. $min(B_k,\ N_k) = N_k$ for any k,

2. $\{B_k\}_{k=1}^{k=K}$ is an overlapping set

In other words, set $\{B_k\}_{k=1}^{K}$ covers 50% or more nodes. Let us denote by $\tilde{k}$, the k for which $min(B_k,\ N_k) = N_k$ (there can be multiple such k). Then $\{B_k\}_{k=1}^{K}$ satisfies

$$\frac{\sum_{k=1}^{K} P_k B_k}{\sum_{k=1}^{K} N_k} \geq \frac{P_{\tilde{k}} B_{\tilde{k}}}{\sum_{k=1}^{K} N_k} \geq \frac{P_{\tilde{k}} N_{\tilde{k}}}{\sum_{k=1}^{K} N_k} \geq \frac{P_K N_K}{\sum_{k=1}^{K} N_k}. \tag{A.37}$$

Similarly, let us assume $B_{k'}$ and $B_{\tilde{k}}$ are overlapping with $\tilde{k} = k' + x$ (there can be multiple overlapping k). Then $\{B_k\}_{k=1}^{K}$ satisfies

$$\frac{\sum_{k=1}^{K} P_k B_k}{\sum_{k=1}^{K} N_k} \geq \frac{P_{\tilde{k}} B_{\tilde{k}} + P_{k'} B_{k'}}{\sum_{k=1}^{K} N_k} \geq \frac{P_{\tilde{k}} N_{\tilde{k}}}{\sum_{k=1}^{K} N_k} \geq \frac{P_K N_K}{\sum_{k=1}^{K} N_k}. \tag{A.38}$$

Observe that, to prove our claim it is sufficient to show that

$$\frac{P_K N_K}{\sum_{k=1}^{K} N_k} \geq 0.5 \Leftrightarrow P_K N_K \geq \frac{N}{2}. \tag{A.39}$$

Using the fact that for a Perfect a-ary tree $P_K = 1$, $N_K = a^K$ and $N = \frac{a(a^K-1)}{a-1}$ the condition (A.39) becomes

$$2 \times a^K \geq \frac{a(a^K - 1)}{a - 1}. \tag{A.40}$$

When $a \geq 2$, we have

$$\begin{aligned} & a \times a^K \geq 2 \times a^K \\ \Leftrightarrow\ & a + a^{K+1} \geq 2 \times a^K \\ \Leftrightarrow\ & 2 \times a^{K+1} - 2 \times a^K \geq a^{K+1} - a \\ \Leftrightarrow\ & 2 \times a^K \geq \frac{a(a^K - 1)}{a - 1}. \end{aligned}$$

Hence, (A.39) holds and this completes our proof.

A.7

We skip the proof of (4.29) and only focus on the proof of (4.30). To show

$$(a+1)^{K+1}[(a)^{K-k+1}-1]-(a)^{K+1}[(a+1)^{K-k+1}-1]\geq 0 \text{ for } a\geq 2$$

is equivalent to show

$$a^{K+1}[(a-1)^{K-k+1}-1]-(a-1)^{K+1}[a^{K-k+1}-1]\geq 0 \text{ for } a\geq 3$$

which can be simplified to

$$(a(a-1))^{K-k+1}[a^k-(a-1)^k]\geq [a^{K+1}-(a-1)^{K+1}]. \tag{A.41}$$

Using binomial expansion, (A.41) becomes

$$\begin{aligned}
&(a(a-1))^{K-k+1}[a^{k-1}+(a-1)a^{k-2}+\cdots+(a-1)^{k-1}]\geq\\
&[a^K+(a-1)a^{K-1}+\cdots+(a-1)^{K-1}a+(a-1)^K]\\
\Leftrightarrow\ &\underbrace{(a-1)^{K-k+1}[a^K+(a-1)a^{K-1}+\cdots+(a-1)^{k-1}a^{K-k+1}]}_{\text{k terms}}\geq\\
&\underbrace{[a^K+(a-1)a^{K-1}+\cdots+(a-1)^{k-1}a^{K-k+1}]}_{\text{k terms}}+\\
&\underbrace{[(a-1)^ka^{K-k}+\cdots+(a-1)^{K-1}a+(a-1)^K]}_{\text{K-k+1 terms}}\\
\Leftrightarrow\ &((a-1)^{K-k+1}-1)[a^K+\cdots+(a-1)^{k-1}a^{K-k+1}]\geq\\
&[(a-1)^ka^{K-k}+\cdots+(a-1)^{K-1}a+(a-1)^K].
\end{aligned} \tag{A.42}$$

Since $a \geq 3$, we have $((a-1)^{K-k+1}-1) \geq (K-k+1) \geq 1$. Hence,

$$((a-1)^{K-k+1}-1)[a^K + \cdots + (a-1)^{k-1}a^{K-k+1}] \geq$$

$$((a-1)^{K-k+1}-1)a^K \geq \underbrace{[(a-1)^k a^{K-k} + \cdots + (a-1)^K]}_{\text{K-k+1 terms}} \tag{A.43}$$

and (A.42) holds.

A.8 Proof of Lemma 4.3.2

To prove the lemma, we first show that any positive deviation $\epsilon \in (0, p]$ in flipping probabilities $(P^k_{1,0}, P^k_{0,1}) = (p, p-\epsilon)$ will result in an increase in D_k. After plugging in $(P^k_{1,0}, P^k_{0,1}) = (p, p-\epsilon)$ in (5.4) and (5.5), we get

$$\pi^k_{1,0} = [\beta^{k-1}_{1,0}(1-P^k_{fa}) + (1-\beta^{k-1}_{0,1})P^k_{fa}] + [\alpha_k(p - P^k_{fa}(2p-\epsilon)) + P^k_{fa}] \tag{A.44}$$

$$\pi^k_{1,1} = [\beta^{k-1}_{1,0}(1-P^k_{d}) + (1-\beta^{k-1}_{0,1})P^k_{d}] + [\alpha_k(p - P^k_{d}(2p-\epsilon)) + P^k_{d}]. \tag{A.45}$$

Now we show that D_k is a monotonically increasing function of the parameter ϵ or in other words, $\frac{dD_k}{d\epsilon} > 0$.

$$\begin{aligned}\frac{dD_k}{d\epsilon} &= \pi^k_{1,0}\left(\frac{\pi^{k'}_{1,0}}{\pi^k_{1,0}} - \frac{\pi^{k'}_{1,1}}{\pi^k_{1,1}}\right) + \pi^{k'}_{1,0}\log\frac{\pi^k_{1,0}}{\pi^k_{1,1}} \\ &+ (1-\pi^k_{1,0})\left(\frac{\pi^{k'}_{1,1}}{1-\pi^k_{1,1}} - \frac{\pi^{k'}_{1,0}}{1-\pi^k_{1,0}}\right) - \pi^{k'}_{1,0}\log\frac{1-\pi^k_{1,0}}{1-\pi^k_{1,1}}\end{aligned} \tag{A.46}$$

where $\frac{d\pi^k_{1,1}}{d\epsilon} = \pi^{k'}_{1,1} = \alpha_k P^k_d$ and $\frac{d\pi^k_{1,0}}{d\epsilon} = \pi^{k'}_{1,0} = \alpha_k P^k_{fa}$. After rearranging the terms in the above equation, the condition $\frac{dD_k}{d\epsilon} > 0$ becomes

$$\frac{1-\pi^k_{1,0}}{1-\pi^k_{1,1}} + \frac{P^k_{fa}}{P^k_d}\log\frac{\pi^k_{1,0}}{\pi^k_{1,1}} > \frac{\pi^k_{1,0}}{\pi^k_{1,1}} + \frac{P^k_{fa}}{P^k_d}\log\frac{1-\pi^k_{1,0}}{1-\pi^k_{1,1}}. \tag{A.47}$$

Since $P_d^k > P_{fa}^k$ and $\beta_{\bar{x},x}^k < 0.5$, $\pi_{1,1}^k > \pi_{1,0}^k$. It can also be proved that $\frac{P_d^k}{P_{fa}^k}\frac{\pi_{1,0}^k}{\pi_{1,1}^k} > 1$. Hence, we have

$$
\begin{aligned}
& 1 + (\pi_{1,0}^k - \pi_{1,1}^k) < \frac{P_d^k}{P_{fa}^k}\frac{\pi_{1,0}^k}{\pi_{1,1}^k} \\
\Leftrightarrow \quad & (\pi_{1,0}^k - \pi_{1,1}^k)[1 + (\pi_{1,0}^k - \pi_{1,1}^k)] > \frac{P_d^k}{P_{fa}^k}\frac{\pi_{1,0}^k}{\pi_{1,1}^k}(\pi_{1,0}^k - \pi_{1,1}^k) \\
\Leftrightarrow \quad & (\pi_{1,0}^k - \pi_{1,1}^k)\left[\frac{1 + (\pi_{1,0}^k - \pi_{1,1}^k)}{\pi_{1,0}^k(1 - \pi_{1,1}^k)}\right] > \frac{P_d^k}{P_{fa}^k}\frac{\pi_{1,0}^k}{\pi_{1,1}^k}\left[\frac{\pi_{1,0}^k - \pi_{1,1}^k}{\pi_{1,0}^k(1 - \pi_{1,1}^k)}\right] \\
\Leftrightarrow \quad & (\pi_{1,0}^k - \pi_{1,1}^k)\left[\frac{1}{1 - \pi_{1,1}^k} + \frac{1}{\pi_{1,0}^k}\right] > \frac{P_d^k}{P_{fa}^k}\left[\frac{\pi_{1,0}^k - \pi_{1,0}^k\pi_{1,1}^k + \pi_{1,0}^k\pi_{1,1}^k - \pi_{1,1}^k}{\pi_{1,1}^k(1 - \pi_{1,1}^k)}\right] \\
\Leftrightarrow \quad & \left[\frac{1 - \pi_{1,1}^k - (1 - \pi_{1,0}^k)}{1 - \pi_{1,1}^k} + \frac{(\pi_{1,0}^k - \pi_{1,1}^k)}{\pi_{1,0}^k}\right] > \frac{P_d^k}{P_{fa}^k}\left[\frac{\pi_{1,0}^k}{\pi_{1,1}^k} - \frac{1 - \pi_{1,0}^k}{1 - \pi_{1,1}^k}\right] \\
\Leftrightarrow \quad & \frac{1 - \pi_{1,0}^k}{1 - \pi_{1,1}^k} + \frac{P_{fa}^k}{P_d^k}\left(1 - \frac{\pi_{1,1}^k}{\pi_{1,0}^k}\right) > \frac{\pi_{1,0}^k}{\pi_{1,1}^k} + \frac{P_{fa}^k}{P_d^k}\left(\frac{1 - \pi_{1,0}^k}{1 - \pi_{1,1}^k} - 1\right). \qquad \text{(A.48)}
\end{aligned}
$$

To prove that (A.47) is true, we apply the logarithm inequality $(x-1) \geq \log x \geq \frac{x-1}{x}$, for $x > 0$ to (A.48). First, let us assume that $x = \frac{\pi_{1,0}^k}{\pi_{1,1}^k}$. Now using the logarithm inequality we can show that $\log\frac{\pi_{1,0}^k}{\pi_{1,1}^k} \geq 1 - \frac{\pi_{1,1}^k}{\pi_{1,0}^k}$. Next, let us assume that $x = \frac{1 - \pi_{1,0}^k}{1 - \pi_{1,1}^k}$. Now using the logarithm inequality it can be shown that $\left[\frac{1 - \pi_{1,0}^k}{1 - \pi_{1,1}^k} - 1\right] \geq \log\frac{1 - \pi_{1,0}^k}{1 - \pi_{1,1}^k}$. Using these results and (A.48), one can prove that condition (A.47) is true.

Similarly, we can show that any non zero deviation $\epsilon \in (0, p]$ in flipping probabilities $(P_{1,0}^k, P_{0,1}^k) = (p - \epsilon, p)$ will result in an increase in D_k, i.e., $\frac{dD_k}{d\epsilon} > 0$, or

$$
\frac{\pi_{1,0}^k}{\pi_{1,1}^k} + \frac{1 - P_{fa}^k}{1 - P_d^k}\log\frac{1 - \pi_{1,0}^k}{1 - \pi_{1,1}^k} > \frac{1 - \pi_{1,0}^k}{1 - \pi_{1,1}^k} + \frac{1 - P_{fa}^k}{1 - P_d^k}\log\frac{\pi_{1,0}^k}{\pi_{1,1}^k}. \qquad \text{(A.49)}
$$

Since $P_d^k > P_{fa}^k$ and $\beta_{\bar{x},x}^k < 0.5$, $\pi_{1,1}^k > \pi_{1,0}^k$. It can also be proved that $\frac{1 - \pi_{1,0}^k}{1 - \pi_{1,1}^k} < \frac{1 - P_{fa}^k}{1 - P_d^k}$.

Hence, we have

$$\frac{1-\pi_{1,0}^k}{1-\pi_{1,1}^k} < \frac{1-P_{fa}^k}{1-P_d^k}\left[1-(\pi_{1,0}^k-\pi_{1,1}^k)\right] \tag{A.50}$$

$$\Leftrightarrow \quad \frac{1-\pi_{1,0}^k}{\pi_{1,1}^k(1-\pi_{1,1}^k)} < \frac{1-P_{fa}^k}{1-P_d^k}\left[\frac{1-(\pi_{1,0}^k-\pi_{1,1}^k)}{\pi_{1,1}^k}\right]$$

$$\Leftrightarrow \quad \frac{1}{\pi_{1,1}^k(1-\pi_{1,1}^k)} < \frac{1-P_{fa}^k}{1-P_d^k}\left[\frac{1-(\pi_{1,0}^k-\pi_{1,1}^k)}{\pi_{1,1}^k(1-\pi_{1,0}^k)}\right]$$

$$\Leftrightarrow \quad \frac{1}{\pi_{1,0}^k-\pi_{1,1}^k}\left[\frac{\pi_{1,0}^k-\pi_{1,0}^k\pi_{1,1}^k+\pi_{1,0}^k\pi_{1,1}^k-\pi_{1,1}^k}{\pi_{1,1}^k(1-\pi_{1,1}^k)}\right] < \frac{1-P_{fa}^k}{1-P_d^k}\left[\frac{1-(\pi_{1,0}^k-\pi_{1,1}^k)}{\pi_{1,1}^k(1-\pi_{1,0}^k)}\right]$$

$$\Leftrightarrow \quad \frac{1}{\pi_{1,0}^k-\pi_{1,1}^k}\left[\frac{\pi_{1,0}^k}{\pi_{1,1}^k}-\frac{1-\pi_{1,0}^k}{1-\pi_{1,1}^k}\right] < \frac{1-P_{fa}^k}{1-P_d^k}\left[\frac{1}{\pi_{1,1}^k}+\frac{1}{1-\pi_{1,0}^k}\right] \tag{A.51}$$

$$\Leftrightarrow \quad \frac{\pi_{1,0}^k}{\pi_{1,1}^k}-\frac{1-\pi_{1,0}^k}{1-\pi_{1,1}^k} > \frac{1-P_{fa}^k}{1-P_d^k}\left[\frac{\pi_{1,0}^k-\pi_{1,1}^k}{\pi_{1,1}^k}+\frac{\pi_{1,0}^k-\pi_{1,1}^k}{1-\pi_{1,0}^k}\right] \tag{A.52}$$

$$\Leftrightarrow \quad \frac{\pi_{1,0}^k}{\pi_{1,1}^k}-\frac{1-\pi_{1,0}^k}{1-\pi_{1,1}^k} > \frac{1-P_{fa}^k}{1-P_d^k}\left[\frac{\pi_{1,0}^k-\pi_{1,1}^k}{\pi_{1,1}^k}+\frac{1-\pi_{1,1}^k-(1-\pi_{1,0}^k)}{1-\pi_{1,0}^k}\right]$$

$$\Leftrightarrow \quad \frac{\pi_{1,0}^k}{\pi_{1,1}^k}+\frac{1-P_{fa}^k}{1-P_d^k}\left[1-\frac{1-\pi_{1,1}^k}{1-\pi_{1,0}^k}\right] > \frac{1-\pi_{1,0}^k}{1-\pi_{1,1}^k}+\frac{1-P_{fa}^k}{1-P_d^k}\left[\frac{\pi_{1,0}^k}{\pi_{1,1}^k}-1\right]. \tag{A.53}$$

To prove that (A.49) is true, we apply the logarithm inequality $(x-1) \geq \log x \geq \dfrac{x-1}{x}$, for $x > 0$ to (A.53). First, let us assume that $x = \dfrac{1-\pi_{1,0}^k}{1-\pi_{1,1}^k}$. Now using the logarithm inequality we can show that $\log \dfrac{1-\pi_{1,0}^k}{1-\pi_{1,1}^k} \geq 1 - \dfrac{1-\pi_{1,1}^k}{1-\pi_{1,0}^k}$. Next, let us assume that $x = \dfrac{\pi_{1,0}^k}{\pi_{1,1}^k}$. Now using the logarithm inequality it can be shown that $\left[\dfrac{\pi_{1,0}^k}{\pi_{1,1}^k}-1\right] \geq \log \dfrac{\pi_{1,0}^k}{\pi_{1,1}^k}$. Using these results and (A.53), one can prove that condition (A.49) is true.

A.9 Proof of Lemma 5.5.2

To prove Lemma 5.5.2, it is sufficient to show that:

1. KLD is a monotonically decreasing function of B_k, and,

2. Attacking parent nodes is a strictly dominant strategy.

Lemma 5.3.4 suggests that the KLD is a monotonically decreasing function of B_k in the region where attacker cannot make $D_k = 0$ and, therefore, (1) is proved. Next, we show that attacking parent nodes is a strictly dominant strategy. In other words, given a cost budget $C_{budget}^{attacker}$, it is more profitable for an attacker to attack the parent nodes. Observe that the KLD at level k is a function of Byzantines' parameter $(B_1, \cdots, B_k)$. Thus, we denote it as $D_k(B_1, \cdots, B_k)$.

In order to prove that attacking parent nodes is a strictly dominant strategy, it is sufficient to show that the attack configuration $S_1 = (B_1, \cdots, B_j, B_{j+1}, \cdots, B_K)$ strictly dominates the attack configuration $S_2 = (B_1, \cdots, B_j - \delta, B_{j+1} + \delta\frac{N_{j+1}}{N_j}, \cdots, B_K)$ for $\delta \in \{1, \cdots, B_j\}$. In other words, we want to show that $P(S_1) > P(S_2)$ and $C(S_1) \leq C(S_2)$. From the cost inequality it follows that $C(S_1) \leq C(S_2)$ because $c_{max} \leq (\min_k N_{k+1}/N_k) \times c_{min} \Rightarrow \tilde{c}_j \leq (N_{j+1}/N_j) \times \tilde{c}_{j+1}$. Also, note that if the attack configuration S_1 strictly dominates the attack configuration S_2, then, it will also strictly dominate any attack configuration $\tilde{S}_2$ with $\tilde{S}_2 = (B_1, \cdots, B_j - \delta, B_{j+1} + \delta\gamma, \cdots, B_K)$, where $\gamma \leq \frac{N_{j+1}}{N_j}$. Next, we show that $P(S_1) > P(S_2)$.

Since $D_j(B_1, \cdots, B_{j-1}, B_j) < D_j(B_1, \cdots, B_{j-1}, B_j - \delta)$, for $\delta \in \{1, \cdots, B_j\},\ \forall j$, it follows that

$$
\begin{aligned}
& D_j(B_1, \cdots, B_{j-1}, B_j) < D_j(B_1, \cdots, B_{j-1}, B_j - \delta) \\
\Leftrightarrow \quad & \sum_{k=1}^{j} D_k(B_1, \cdots, B_k) < \sum_{k=1}^{j-1} D_k(B_1, \cdots, B_k) + D_j(B_1, \cdots, B_{j-1}, B_j - \delta) \\
\Leftrightarrow \quad & \sum_{k=1}^{K} D_k(B_1, \cdots, B_k) < \sum_{k=1}^{j-1} D_k(B_1, \cdots, B_k) + D_j(B_1, \cdots, B_{j-1}, B_j - \delta) \\
& \qquad + \sum_{k=j+1}^{K} D_k(B_1, \cdots, B_j - \delta, B_{j+1} + \delta\frac{N_{j+1}}{N_j}, B_{j+2}, \cdots, B_k),
\end{aligned}
$$

where the last inequality follows from the fact that $\frac{B_j}{N_j} + \frac{B_{j+1}}{N_{j+1}} = \frac{B_j - \delta}{N_j} + \frac{B_{j+1} + \frac{N_{j+1}}{N_j}\delta}{N_{j+1}}$ and, therefore,

$$D_k(B_1, \cdots, B_j, B_{j+1}, \cdots, B_k) = D_k(B_1, \cdots, B_j - \delta, B_{j+1} + \frac{N_{j+1}}{N_j}\delta, \cdots, B_k).$$

This implies that the set S_1 strictly dominates the set S_2. From the results in Lemma 5.3.4, it is seen that the profit is an increasing function of the attack nodes. Lemma 5.3.4 in conjunction with the fact that attacking parent nodes is a strictly dominant strategy implies Lemma 5.5.2.

A.10

To prove the proposition, we start with the inequality (6.10)

$$d^k_{min} \geq \frac{2(M-2)}{[1 - 4q^k_{max}(1 - q^k_{max})] - (1/a_k)[(2/q^k_{max}) - 2]} \tag{A.54}$$

$$\implies \frac{d^k_{min}}{2}\left(1 - 4q^k_{max}(1 - q^k_{max})\right) \geq (M-2) + \frac{d^k_{min}}{a_k}\left(\frac{1}{q^k_{max}} - 1\right) \tag{A.55}$$

$$\implies \frac{d^k_{min}}{2}\log\left(\frac{1}{4q^k_{max}(1 - q^k_{max}}\right) \geq \log(M-1) + \frac{d^k_{min}}{a_k}\log\frac{1}{q^k_{max}} \tag{A.56}$$

$$\implies \left[q^k_{max}\right]^{\frac{d^k_{min}}{a_k}} \geq (M-1)\left[\sqrt{4q^k_{max}(1 - q^k_{max})}\right]^{d^k_{min}} \tag{A.57}$$

where (A.55) is true because a_k satisfies (6.11), and (A.56) can be proved by applying the logarithm inequality: $(x-1) \geq \log x \geq \frac{x-1}{x}$, for $x > 0$. Now, for $k = 1, \cdots, K$

$$
\begin{aligned}
& Q_m^{k-1} = Pr(\text{decision at level } k-1 \neq H_m \mid H_m) \\
\leq\ & Pr(d^k(\mathbf{u^k}, \mathbf{c_m^k}) \geq \min_{1\leq l\leq M,\, l\neq m} d^k(\mathbf{u^k}, \mathbf{c_l^k}) \mid H_m) \\
\leq\ & \sum_{\substack{l=1 \\ l\neq m}}^{M} Pr(d^k(\mathbf{u^k}, \mathbf{c_m^k}) \geq d^k(\mathbf{u^k}, \mathbf{c_l^k}) \mid H_m) \\
\leq\ & \sum_{\substack{l=1 \\ l\neq m}}^{M} \left[\sqrt{4Q_{mm}^k(1-Q_{mm}^k)}\right]^{d^k(\mathbf{c_l^k}, \mathbf{c_m^k})} \qquad \text{(A.58)} \\
\leq\ & (M-1)\left[\sqrt{4q_{max}^k(1-q_{max}^k)}\right]^{d_{min}^k} \qquad \text{(A.59)} \\
\leq\ & \left[q_{max}^k\right]^{\frac{d_{min}^k}{a_k}}. \qquad \text{(A.60)}
\end{aligned}
$$

Note that (A.58) is true when $Q_m^k < \frac{1}{2}$ as shown in [18], which holds when (6.10) and (6.11) are true. Using the above results, the average probability of error can be bounded as follows.

$$
\begin{aligned}
P_e^0 &= \sum_{m=1}^{M} P_m Pr(\text{decision at the FC} \neq H_m \mid H_m) \\
&\leq \sum_{m=1}^{M} P_m \left[q_{max}^1\right]^{\frac{d_{min}^1}{a_1}} \\
&= \left[q_{max}^1\right]^{\frac{d_{min}^1}{a_1}}
\end{aligned}
$$

Now, since $Q_m^1 \leq [q_{max}^2]^{\frac{d_{min}^2}{a_2}}\ \forall m$, we have $q_{max}^1 \leq [q_{max}^2]^{\frac{d_{min}^2}{a_2}}$. Continuing in this manner, we get

$$
P_e^0 \leq \left[q_{max}^K\right]^{\prod_{k=1}^{K} \frac{d_{min}^k}{a_k}}.
$$

A.11

To prove the proposition, we first establish the following set of inequalities (Please see (A.58)-(A.60))

$$
\begin{aligned}
& \sum_{m=1}^{M} P_m^k Pr(\text{decision at level } k-1 \neq H_m^k \mid H_m^k) \\
\leq & \sum_{m=1}^{M} P_m^k Pr(d^k(\mathbf{u^k}, \mathbf{c_m^k}) \geq \min_{1 \leq l \leq M,\, l \neq m} d^k(\mathbf{u^k}, \mathbf{c_l^k}) \mid H_m^k) \\
\leq & \sum_{m=1}^{M} P_m^k \sum_{\substack{l=1 \\ l \neq m}}^{M} Pr(d^k(\mathbf{u^k}, \mathbf{c_m^k}) \geq d^k(\mathbf{u^k}, \mathbf{c_l^k}) \mid H_m^k) \\
\leq & \sum_{m=1}^{M} P_m^k \sum_{\substack{l=1 \\ l \neq m}}^{M} \left[\sqrt{4Q_{mm}^k(1-Q_{mm}^k)}\right]^{d^k(\mathbf{c_l^k}, \mathbf{c_m^k})} \\
\leq & (M-1)\left[\sqrt{4q_{max}^k(1-q_{max}^k)}\right]^{d_{min}^k}.
\end{aligned}
$$

Therefore,

$$
\begin{aligned}
& \sum_{l=1}^{M} P_m^k Pr(\text{decision at level } k-1 \neq H_m^k \mid H_m^k) \\
& \qquad \leq (M-1)\left[\sqrt{4q_{max}^k(1-q_{max}^k)}\right]^{d_{min}^k} \\
\Leftrightarrow & \sum_{l=1}^{M} P_m^k Pr(\text{decision at level } k-1 = H_m^k \mid H_m^k) \\
& \qquad \geq 1-(M-1)\left[\sqrt{4q_{max}^k(1-q_{max}^k)}\right]^{d_{min}^k}.
\end{aligned}
$$

Now,

$$\begin{aligned} P_e^0 &= 1 - \prod_{k=1}^{K} \sum_{l=1}^{M} P_m^k Pr(\text{decision at level } k-1 = H_m^k \mid H_m^k) \\ &\leq 1 - \prod_{k=1}^{K} \left[1 - (M-1)(4q_{max}^k(1-q_{max}^k))^{\frac{d_{min}^k}{2}} \right]. \end{aligned}$$

A.12

The local test statistic Y_i has the mean

$$mean_i = \begin{cases} M\sigma_i^2 & \text{if } H_0 \\ (M+\eta_i)\sigma_i^2 & \text{if } H_1 \end{cases}$$

and the variance

$$Var_i = \begin{cases} 2M\sigma_i^4 & \text{if } H_0 \\ 2(M+2\eta_i)\sigma_i^4 & \text{if } H_1. \end{cases}$$

The goal of Byzantine nodes is to make the deflection coefficient as small as possible. Since the Deflection Coefficient is always non-negative, the Byzantines seek to make $\mathcal{D}(\Lambda) = \frac{(\mu_1 - \mu_0)^2}{\sigma_{(0)}^2} = 0$. The conditional mean $\mu_k = \mathbb{E}[\Lambda|H_k]$ and conditional variance $\sigma_{(0)}^2 = \mathbb{E}[(\Lambda - \mu_0)^2|H_0]$ of the global test statistic, $\Lambda = (\sum_{i=1}^{N_1} \tilde{w}_i \tilde{Y}_i + \sum_{i=N_1+1}^{N} w_i Y_i)/(\sum w)$, can be computed and are given by (7.3), (7.4) and (7.5), respectively. After substituting values from (7.3), (7.4) and (7.5), the condition to make $\mathcal{D}(\Lambda) = 0$ becomes

$$\sum_{i=1}^{N_1} \tilde{w}_i (2P_i\Delta_i - \eta_i\sigma_i^2) = \sum_{i=N_1+1}^{N} w_i\eta_i\sigma_i^2$$

A.13

Note that, for sufficiently large M (in practice $M \geq 12$), the distribution of Byzantine's data $\tilde{Y_i}$ given H_k is a Gaussian mixture which comes from $\mathcal{N}((\mu_{1k})_i, (\sigma_{1k})_i^2)$ with probability $(1-P)$ and from $\mathcal{N}((\mu_{2k})_i, (\sigma_{2k})_i^2)$ with probability P, and

$$(\mu_{10})_i = M\sigma_i^2,\ (\mu_{20})_i = M\sigma_i^2 + \Delta_i$$

$$(\mu_{11})_i = (M+\eta_i)\sigma_i^2,\ (\mu_{21})_i = (M+\eta_i)\sigma_i^2 - \Delta_i$$

$$(\sigma_{10})_i^2 = (\sigma_{20})_i^2 = 2M\sigma_i^4,\ \textit{and}\ (\sigma_{11})_i^2 = (\sigma_{21})_i^2 = 2(M+2\eta_i)\sigma_i^4.$$

Now, the probability density function (PDF) of $x_{ji}^t = w_{ji}^t \tilde{Y_i}$ conditioned on H_k can be derived as

$$f(x_{ji}^t|H_k) = (1-P)\phi(w_{ji}^t(\mu_{1k})_i, (w_{ji}^t(\sigma_{1k})_i)^2)$$

$$+ P\phi(w_{ji}^t(\mu_{2k})_i, (w_{ji}^t(\sigma_{2k})_i)^2) \tag{A.61}$$

where $\phi(x|\mu,\sigma^2)$ (for notational convenience denoted as $\phi(\mu,\sigma^2)$) is the PDF of $X \sim \mathcal{N}(\mu,\sigma^2)$ and $\phi(x|\mu,\sigma^2) = \frac{1}{\sigma\sqrt{2\pi}} e^{-(x-\mu)^2/2\sigma^2}$.

Now, for the three node case, the transient test statistic $\tilde{\Lambda}_j^t = w_{j1}^t\tilde{Y_1} + w_{j2}^t\tilde{Y_2} + w_{j3}^t Y_3$, is a summation of independent random variables. The conditional PDF of $X_{ji}^t = w_{ji}^t\tilde{Y_i}$ is given in (A.61). Notice that, PDF of $\tilde{\Lambda}_j^t$ is the convolution $(*)$ of $f(x_{j1}^t) = (1-P)\phi(\mu_1^1, (\sigma_1^1)^2) + P\phi(\mu_1^2, (\sigma_1^2)^2)$, $f(x_{j2}^t) = (1-P)\phi(\mu_2^1, (\sigma_2^1)^2) + P\phi(\mu_2^2, (\sigma_2^2)^2))$ and $f(x_{j3}^t) = \phi(\mu_3^1, (\sigma_3^1)^2)$.

$$f(z_j^t) = f(x_{j1}^t) * f(x_{j2}^t) * f(x_{j3}^t)$$

$$f(z_j^t) = [(1-P)\phi(\mu_1^1, (\sigma_1^1)^2) + P\phi(\mu_1^2, (\sigma_1^2)^2)]*$$

$$[(1-P)\phi(\mu_2^1, (\sigma_2^1)^2) + P\phi(\mu_2^2, (\sigma_2^2)^2)] * \phi(\mu_3^1, (\sigma_3^1)^2)$$

$$= (1-P)^2[\phi(\mu_1^1,(\sigma_1^1)^2) * \phi(\mu_2^1,(\sigma_2^1)^2) * \phi(\mu_3^1,(\sigma_3^1)^2)]$$

$$+(P)^2[\phi(\mu_1^2,(\sigma_1^2)^2) * \phi(\mu_2^2,(\sigma_2^2)^2)) * \phi(\mu_3^1,(\sigma_3^1)^2)]$$

$$+P(1-P)[\phi(\mu_1^2,(\sigma_1^2)^2) * \phi(\mu_2^1,(\sigma_2^1)^2) * \phi(\mu_3^1,(\sigma_3^1)^2)]$$

$$+(1-P)P[\phi(\mu_1^1,(\sigma_1^1)^2) * \phi(\mu_2^2,(\sigma_2^2)^2) * \phi(\mu_3^1,(\sigma_3^1)^2)]$$

Now, using the fact that convolution of two normal PDFs $\phi(\mu_i,\sigma_i^2)$ and $\phi(\mu_j,\sigma_j^2)$ is again normally distributed with mean $(\mu_i+\mu_j)$ and variance $(\sigma_i^2+\sigma_j^2)$, we can derive the results below.

$$f(z_j^t) = (1-P)^2[\phi(\mu_1^1+\mu_2^1+\mu_3^1,(\sigma_1^1)^2+(\sigma_2^1)^2+(\sigma_3^1)^2)]$$

$$+P^2[\phi(\mu_1^2+\mu_2^2+\mu_3^1,(\sigma_1^2)^2+(\sigma_2^2)^2+(\sigma_3^1)^2)]$$

$$+P(1-P)[\phi(\mu_1^2+\mu_2^1+\mu_3^1,(\sigma_1^2)^2+(\sigma_2^1)^2+(\sigma_3^1)^2)]$$

$$+(1-P)P[\phi(\mu_1^1+\mu_2^2+\mu_3^1,(\sigma_1^1)^2+(\sigma_2^2)^2+(\sigma_3^1)^2)].$$

REFERENCES

[1] S. Agrawal and S. Vishwanath, "Secrecy using compressive sensing," in *IEEE Information Theory Workshop (ITW).*, Oct 2011, pp. 563–567.

[2] S. Alhakeem and P. K. Varshney, "A unified approach to the design of decentralized detection systems," *IEEE Trans. Aerosp. Electron. Syst.*, vol. 31, no. 1, pp. 9–20, Jan. 1995.

[3] Aliance, "Z. Zigbee Specifications," *Zigbee Standard Organisation*, 2008.

[4] S. Appadwedula, V. V. Veeravalli, and D. L. Jones, "Energy-efficient detection in sensor networks," *IEEE J. Sel. Areas Commun.*, vol. 23, no. 4, pp. 693–702, Apr. 2005.

[5] ——, "Decentralized detection with censoring sensors," *IEEE Trans. Signal Process.*, vol. 56, no. 4, pp. 1362–1373, Apr. 2008.

[6] W. Baek and S. Bommareddy, "Optimal M-ary data fusion with distributed sensors," *IEEE Trans. Aersop. Electron. Syst.*, vol. 31, no. 3, Jul 1995.

[7] H. Bai, Z. Zhu, G. Li, and S. Li, "Design of Optimal Measurement Matrix for Compressive Detection," in *Proceedings of the Tenth International Symposium on Wireless Communication Systems (ISWCS 2013).*, Aug 2013, pp. 1–5.

[8] J. Bard, "Some properties of the bilevel programming problem," *Journal of Optimization Theory and Applications*, vol. 68, no. 2, pp. 371–378, 1991. [Online]. Available: http://dx.doi.org/10.1007/BF00941574

[9] R. S. Blum, S. A. Kassam, and H. V. Poor, "Distributed detection with multiple sensors: Part II – Advanced topics," *Proc. IEEE*, vol. 85, no. 1, pp. 64–79, Jan. 1997.

[10] J. Cao and Z. Lin, "Bayesian signal detection with compressed measurements," *Information Sciences*, vol. 289, no. 0, pp. 241 – 253, 2014.

[11] P. Casazza and M. Leon, "Existence And Construction Of Finite Tight Frames," *J. Concr. Appl. Math*, 2006.

[12] J.-F. Chamberland and V. V. Veeravalli, "How dense should a sensor network be for detection with correlated observations?" *IEEE Trans. Inf. Theory*, vol. 52, no. 11, pp. 5099–5106, Nov. 2006.

[13] ——, "Wireless sensors in distributed detection applications," *IEEE Signal Process. Mag.*, vol. 24, no. 3, pp. 16–25, May 2007.

[14] B. Chen and P. K. Willett, "On the Optimality of the Likelihood-ratio Test for Local Sensor Decision Rules in the Presence of Nonideal Channels," *IEEE Trans. Inf. Theor.*, vol. 51, no. 2, pp. 693–699, Feb. 2005.

[15] B. Chen, L. Tong, and P. K. Varshney, "Channel-aware distributed detection in wireless sensor networks," *IEEE Signal Process. Mag.*, vol. 23, no. 4, pp. 16–26, Jul. 2006.

[16] ——, "Channel-aware distributed detection in wireless sensor networks," *IEEE Signal Process. Mag. (Special Issue on Distributed Signal Processing for Sensor Networks)*, vol. 23, pp. 16–26, Jul. 2006.

[17] H. Chen and P. K. Varshney, "Performance limit for distributed estimation systems with identical one-bit quantizers," *IEEE Trans. Signal Process.*, vol. 58, no. 1, pp. 1607–1621, Jan. 2009.

[18] P.-N. Chen, T.-Y. Wang, Y. S. Han, P. K. Varshney, and C. Yao, "Asymptotic performance analysis for minimum Hamming distance fusion [wireless sensor network applications]," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process. (ICASSP 2005)*, vol. 4, Mar. 2005, pp. 865–868.

[19] R. Chen, J.-M. Park, and K. Bian, "Robust distributed spectrum sensing in cognitive radio networks," in *Proc. 27th Conf. Comput. Commun., Phoenix, AZ*, 2008, pp. 1876–1884.

[20] H. Chernoff, "A measure of asymptotic efficiency for tests of a hypothesis based on the sum of observations," *The Annals of Mathematical Statistics*, vol. 23, pp. 493–507, December 1952.

[21] J. Choi, J. Ha, and H. Jeon, "Physical layer security for wireless sensor networks," in *IEEE 24th International Symposium on Personal Indoor and Mobile Radio Communications (PIMRC).*, Sept 2013, pp. 1–6.

[22] T. Clouqueur, K. Saluja, and P. Ramanathan, "Fault tolerance in collaborative sensor networks for target detection," *Computers, IEEE Transactions on*, vol. 53, no. 3, pp. 320–333, Mar 2004.

[23] T. Cover and J. Thomas, *Elements of Information Theory*. New York:Wiley, 1991.

[24] O. Dabeer and E. Masry, "Multivariate signal parameter estimation under dependent noise from 1-bit dithered quantized data," *IEEE Trans. Inf. Theory*, vol. 54, no. 4, pp. 1637–1654, 2008.

[25] M. Davenport, P. Boufounos, M. Wakin, and R. Baraniuk, "Signal processing with compressive measurements," *Selected Topics in Signal Processing, IEEE Journal of*, vol. 4, no. 2, pp. 445–460, April 2010.

[26] M. A. Davenport, M. F. Duarte, M. B. Wakin, J. N. Laskar, D. Takhar, K. F. Kelly, and R. G. Baraniukr, "The Smashed Filter for Compressive Classification and Target Recognition," in *Storage and Retrieval for Image and Video Databases*, Submitted.

[27] V. G. Deineko and G. J. Woeginger, "A well-solvable special case of the bounded knapsack problem," *Operations Research Letters*, vol. 39, pp. 118–120, 2011.

[28] F. F. Digham, M. S. Alouini, and M. K. Simon, "On the energy detection of unknown signals over fading channels," in *IEEE International Conference on Communications, 2003. ICC '03.*, vol. 5, May 2003, pp. 3575–3579 vol.5.

[29] M. Duarte, M. Davenport, M. Wakin, and R. Baraniuk, "Sparse Signal Detection from Incoherent Projections," in *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP).*, vol. 3, May 2006, pp. III–III.

[30] G. Ferrari, M. Martalo, and R. Pagliari, "Decentralized Detection in Clustered Sensor Networks," *Aerospace and Electronic Systems, IEEE Transactions on*, vol. 47, no. 2, pp. 959–973, April 2011.

[31] J. Font-Segura, G. Vazquez, and J. Riba, "Asymptotic error exponents in energy-detector and estimator-correlator signal detection," in *IEEE International Conference on Communications (ICC).*, June 2012, pp. 3676–3680.

[32] M. Gagrani, P. Sharma, S. Iyengar, V. S. S. Nadendla, A. Vempaty, H. Chen, and P. K. Varshney, "On noise-enhanced distributed inference in the presence of Byzantines," in *Proc. 49th Annu. Allerton Conf. Commun. Control Comput.*, Sep. 2011, pp. 1222–1229.

[33] S. Goel and R. Negi, "Guaranteeing secrecy using artificial noise," *IEEE Transactions on Wireless Communications*, vol. 7, no. 6, pp. 2180–2189, June 2008.

[34] O. Gurewitz, A. de Baynast, and E. W. Knightly, "Cooperative Strategies and Achievable Rate for Tree Networks With Optimal Spatial Reuse," *IEEE Trans. Inf. Theor.*, vol. 53, no. 10, pp. 3596–3614, Oct. 2007. [Online]. Available: http://dx.doi.org/10.1109/TIT.2007.905000

[35] J. Haupt, R. Castro, R. Nowak, G. Fudge, and A. Yeh, "Compressive Sampling for Signal Classification," in *Fortieth Asilomar Conference on Signals, Systems and Computers.*, Oct 2006, pp. 1430–1434.

[36] J. Haupt and R. Nowak, "Compressive sampling for signal detection," in *Acoustics, Speech and Signal Processing, 2007. ICASSP 2007. IEEE International Conference on*, vol. 3, April 2007, pp. III–1509–III–1512.

[37] X. He, H. Dai, and P. Ning, "HMM-based malicious user detection for robust collaborative spectrum sensing," *IEEE J. Sel. Areas Commun.*, vol. 31, no. 11, pp. 2196–2208, Nov. 2013.

[38] R. A. Horn and C. R. Johnson, *Matrix Analysis*. Cambridge, U.K.: Cambridge Univ. Press, 1987.

[39] Y.-H. Hu and D. Li, "Energy based collaborative source localization using acoustic micro-sensor array," in *2002 IEEE Workshop on Multimedia Signal Processing*, Dec 2002, pp. 371–375.

[40] IEEE. P802.22b Draft Standard for Wireless Regional Area Networks (WRAN)–Specific requirements Part 22: Cognitive Wireless RAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications: Policies and Procedures for Operation in the TV Bands Amendment: Enhancement for Broadband Services and Monitoring Applications.

[41] S. Iyengar, P. K. Varshney, and T. Damarla, "A parametric copula based framework for hypotheses testing using heterogeneous data," *IEEE Trans. Signal Process.*, vol. 59, no. 5, pp. 2308–2319, May 2011.

[42] S. Jafarizadeh, "Fastest Distributed Consensus Averaging Problem on Perfect and Complete n-ary Tree networks," *CoRR*, vol. abs/1005.2662, 2010.

[43] R. Jiang and B. Chen, "Fusion of censored decisions in wireless sensor networks," *IEEE Trans. Wireless Commun.*, vol. 4, no. 6, pp. 2668–2673, Nov. 2005.

[44] B. Kailkhura, V. S. S. Nadendla, and P. K. Varshney, "Distributed inference in the presence of eavesdroppers: a survey," *IEEE Communications Magazine*, vol. 53, no. 6, pp. 40–46, June 2015.

[45] B. Kailkhura, T. Wimalajeewa, and P. K. Varshney, "On Physical Layer Secrecy of Collaborative Compressive Detection," in *Forty Eighth Asilomar Conf. on Signals, Systems and Computers*, Nov 2014.

[46] B. Kailkhura, S. Brahma, Y. S. Han, and P. K. Varshney, "Optimal distributed detection in the presence of byzantines," in *Proc. The 38th International Conference on Acoustics, Speech, and Signal Processing (ICASSP 2013)*, Vancouver, Canada, May 2013.

[47] ——, "Distributed Detection in Tree Topologies With Byzantines," *IEEE Trans. Signal Process.*, vol. 62, pp. 3208–3219, June 2014.

[48] B. Kailkhura, S. Brahma, and P. K. Varshney, "Optimal byzantine attack on distributed detection in tree based topologies," in *Proc. International Conference on Computing, Networking and Communications Workshops (ICNC-2013)*, San Diego, CA, January 2013, pp. 227–231.

[49] B. Kailkhura, S. K. Brahma, Y. S. Han, and P. K. Varshney, "Distributed detection in tree topologies with Byzantines," *IEEE Trans. Signal Process.*, vol. 62, no. 12, pp. 3208–3219, Jun. 15 2014.

[50] B. Kailkhura, Y. S. Han, S. K. Brahma, and P. K. Varshney, "On covert data falsification attacks on distributed detection systems," in *Proc. IEEE 13th Int. Symp. Commun. Inf. Tech. (ISCIT 2013)*, Sep. 2013, pp. 412–417.

[51] ——, "Asymptotic analysis of distributed Bayesian detection with Byzantine data," *IEEE Signal Process. Lett.*, vol. 22, no. 5, pp. 608–612, May 2015.

[52] S. Kar, H. Chen, and P. K. Varshney., "Optimal identical binary quantizer design for distributed estimation," *IEEE Trans. Signal Process.*, vol. 60, no. 7, pp. 3896–3901, Jul. 2012.

[53] S. Karande and H. Radha, "Influence of Graph Properties of Peer-to-Peer Topologies on Video Streaming with Network Channel Coding," in *Multimedia and Expo, 2006 IEEE International Conference on*, 2006, pp. 825–828.

[54] S. M. Kay, *Fundamentals of Statistical Signal Processing, Volume 2: Detection Theory*. ser. Prentice Hall Signal Processing Series, A. V. Oppenheim, Ed. Prentice Hall PTR, 1998.

[55] K. Krishnamurthy, M. Raginsky, and R. Willett, "Hyperspectral target detection from incoherent projections," in *IEEE International Conference on Acoustics Speech and Signal Processing (ICASSP).*, March 2010, pp. 3550–3553.

[56] S. Kullback, *Information Theory and Statistics*, 1968.

[57] W.-M. Lam and A. R. Reibman, "Design of quantizers for decentralized estimation systems," *IEEE Trans. Comput.*, vol. 41, no. 11, pp. 1602–1605, Nov. 1993.

[58] L. Lamport, R. Shostak, and M. Pease, "The byzantine generals problem," *ACM Trans. Program. Lang. Syst.*, vol. 4, no. 3, pp. 382–401, Jul. 1982. [Online]. Available: http://doi.acm.org/10.1145/357172.357176

[59] Z. Li, F. Yu, and M. Huang, "A Distributed Consensus-Based Cooperative Spectrum-Sensing Scheme in Cognitive Radios," *Vehicular Technology, IEEE Transactions on*, vol. 59, no. 1, pp. 383–393, Jan 2010.

[60] Y. Lin, B. Chen, and P. Varshney, "Decision fusion rules in multi-hop wireless sensor networks," *IEEE Trans. Aerosp. Electron. Syst.*, vol. 41, no. 2, pp. 475 – 488, april 2005.

[61] S. Liu, M. Chen, S. Sengupta, M. Chiang, J. Li, and P. Chou, "P2P Streaming Capacity under Node Degree Bound," in *Distributed Computing Systems (ICDCS), 2010 IEEE 30th International Conference on*, 2010, pp. 587–598.

[62] S. Liu, H. Zhu, S. Li, X. Li, C. Chen, and X. Guan, "An adaptive deviation-tolerant secure scheme for distributed cooperative spectrum sensing," in *Global Communications Conference (GLOBECOM), 2012 IEEE*, Dec 2012, pp. 603–608.

[63] S. Lloyd, "Least squares quantization in PCM," *IEEE Trans. Inf. Theory*, vol. 28, no. 2, pp. 129–137, Mar. 1982.

[64] X. Luo, M. Dong, and Y. Huang, "On distributed fault-tolerant detection in wireless sensor networks," *Computers, IEEE Transactions on*, vol. 55, no. 1, pp. 58–70, Jan 2006.

[65] Z.-Q. Luo, "Universal decentralized estimation in a bandwidth constrained sensor network," *IEEE Trans. Inf. Theory*, vol. 51, no. 6, pp. 2210–2219, Jun. 2005.

[66] S. Marano, V. Matta, and L. Tong, "Distributed detection in the presence of byzantine attacks," *IEEE Trans. Signal Process.*, vol. 57, no. 1, pp. 16 –29, Jan. 2009.

[67] J. Max, "Quantizing for minimum distortion," *IRE Trans. Inf. Theory*, vol. 6, no. 1, pp. 7–12, Mar. 1960.

[68] J. Meng, H. Li, and Z. Han, "Sparse event detection in wireless sensor networks using compressive sensing," in *43rd Annual Conference on Information Sciences and Systems (CISS).*, March 2009, pp. 181–185.

[69] A. Min, K.-H. Kim, and K. Shin, "Robust cooperative sensing via state estimation in cognitive radio networks," in *New Frontiers in Dynamic Spectrum Access Networks (DySPAN), 2011 IEEE Symposium on*, May 2011, pp. 185–196.

[70] V. S. S. Nadendla, Y. S. Han, and P. K. Varshney, "Distributed inference with m-ary quantized data in the presence of Byzantine attacks," *IEEE Trans. Signal Process.*, vol. 62, no. 10, pp. 2681–2695, May 15 2014.

[71] M. Naraghi-Pour and V. Nadendla, "Secure detection in wireless sensor networks using a simple encryption method," in *Wireless Communications and Networking Conference (WCNC), 2011 IEEE*, March 2011, pp. 114 –119.

[72] R. Negi and S. Goel, "Secret communication using artificial noise," in *VTC-2005-Fall. 2005 IEEE 62nd Vehicular Technology Conference, 2005.*, vol. 3, Sept 2005, pp. 1906–1910.

[73] R. Niu and P. K. Varshney, "Target location estimation in sensor networks with quantized data," *IEEE Trans. Signal Process.*, vol. 54, no. 12, pp. 4519–4528, Dec. 2006.

[74] T. Ogasawara and M. Takahashi, "Independence of quadratic quantities in a normal system." *J. Sci. Hiroshima University*, vol. 15, pp. 1–9, 1951.

[75] R. Olfati-Saber, J. Fax, and R. Murray, "Consensus and Cooperation in Networked Multi-Agent Systems," *Proceedings of the IEEE*, vol. 95, no. 1, pp. 215–233, Jan 2007.

[76] O. Ozdemir, R. Niu, and P. K. Varshney, "Channel aware target localization with quantized data in wireless sensor networks," *IEEE Trans. Signal Process.*, vol. 57, no. 3, pp. 1190–1202, Mar. 2009.

[77] V. N. Padmanabhan, H. J. Wang, P. A. Chou, and K. Sripanidkulchai, "Distributing streaming media content using cooperative networking," in *Proc. International Workshop on Network and Operating Systems Support for Digital Audio and Video*, ser. NOSSDAV '02. New York, NY, USA: ACM, 2002, pp. 177–186. [Online]. Available: http://doi.acm.org/10.1145/507670.507695

[78] F. Pasqualetti, A. Bicchi, and F. Bullo, "Consensus Computation in Unreliable Networks: A System Theoretic Approach," *Automatic Control, IEEE Transactions on*, vol. 57, no. 1, pp. 90–104, Jan 2012.

[79] H. V. Poor, *An Introduction to Signal Detection and Estimation (2Nd Ed.)*. New York, NY, USA: Springer-Verlag New York, Inc., 1994.

[80] W. H. Press, S. A. Teukolsky, W. T. Vetterling, and B. P. Flannery, *Numerical Recipes: The Art of Scientific Computing*, 3rd ed. Cambridge University Press, 2007.

[81] Z. Quan, S. Cui, and A. Sayed, "Optimal Linear Cooperation for Spectrum Sensing in Cognitive Radio Networks," *Selected Topics in Signal Processing, IEEE Journal of*, vol. 2, no. 1, pp. 28–40, Feb 2008.

[82] Y. Rachlin and D. Baron, "The secrecy of compressed sensing measurements," in *46th Annual Allerton Conference on Communication, Control, and Computing.*, Sept 2008, pp. 813–817.

[83] C. Rago, P. Willett, and Y. Bar-Shalom, "Censoring sensors: A low-communication-rate scheme for distributed detection," *IEEE Trans. Aerosp. Electron. Syst.*, vol. 32, no. 2, pp. 554–568, Apr. 1996.

[84] M. Ramezani Mayiami, B. Seyfe, and H. Bafghi, "Perfect secrecy via compressed sensing," in *Iran Workshop on Communication and Information Theory (IWCIT).*, May 2013, pp. 1–5.

[85] A. Rawat, P. Anand, H. Chen, and P. Varshney, "Collaborative spectrum sensing in the presence of byzantine attacks in cognitive radio networks," *IEEE Trans. Signal Process.*, vol. 59, no. 2, pp. 774 –786, Feb 2011.

[86] A. Ribeiro and G. Giannakis, "Bandwidth-constrained distributed estimation for wireless sensor Networks-part I: Gaussian case," *IEEE Trans. Signal Process.*, vol. 54, no. 3, pp. 1131–1143, Mar. 2006.

[87] ——, "Bandwidth-constrained distributed estimation for wireless sensor networks-part II: unknown probability density function," *IEEE Trans. Signal Process.*, vol. 54, no. 7, pp. 2784–2796, Jul. 2006.

[88] U. Rogers and H. Chen, "Heterogeneous Sensor Networks with convex constraints," in *IEEE Aerospace Conference.*, March 2013, pp. 1–10.

[89] C. E. Shannon, *Reliable Machines from Unreliable Components*, MIT, Cambridge, MA, Mar. 1956, notes of first five lectures in the seminar of information theory.

[90] X. Sheng and Y.-H. Hu, "Maximum likelihood multiple-source localization using acoustic energy measurements with wireless sensor networks," *Signal Processing, IEEE Transactions on*, vol. 53, no. 1, pp. 44–53, Jan 2005.

[91] J. Sherman and W. J. Morrison, "Adjustment of an inverse matrix corresponding to a change in one element of a given matrix," *The Annals of Mathematical Statistics*, vol. 21, no. 1, pp. 124–127, 03 1950.

[92] W. Shi, T. W. Sun, and R. D. Wesel, "Optimal binary distributed detection," in *Proc. The 33rd Asilomar Conference on Signals, Systems, and Computers*, 1999, pp. 24–27.

[93] E. Soltanmohammadi, M. Orooji, and M. Naraghi-Pour, "Decentralized hypothesis testing in wireless sensor networks in the presence of misbehaving nodes," *IEEE Trans. Inf. Forensics Security*, vol. 8, no. 1, pp. 205–215, 2013.

[94] S. Sundaram and C. Hadjicostis, "Distributed Function Calculation via Linear Iterative Strategies in the Presence of Malicious Agents," *Automatic Control, IEEE Transactions on*, vol. 56, no. 7, pp. 1495–1508, July 2011.

[95] A. Sundaresan, P. K. Varshney, and N. S. V. Rao, "Copula-based fusion of correlated decisions," *IEEE Trans. Aerosp. Electron. Syst.*, vol. 47, no. 1, pp. 454–471, Jan. 2011.

[96] P. E. Swaszek, "On the performance of serial networks in distributed detection," *IEEE Trans. Aerosp. Electron. Syst.*, vol. 29, no. 1, pp. 254–260, Jan. 1993.

[97] H. Tang, F. Yu, M. Huang, and Z. Li, "Distributed consensus-based security mechanisms in cognitive radio mobile ad hoc networks," *Communications, IET*, vol. 6, no. 8, pp. 974–983, May 2012.

[98] W. P. Tay, "Decentralized Detection in Resource-limited Sensor Network Architectures," Ph.D. dissertation, Massachusetts Institute of Technology, Feb. 2008.

[99] W. P. Tay, J. Tsitsiklis, and M. Win, "Data Fusion Trees for Detection: Does Architecture Matter?" *IEEE Trans. Inf. Theory*, vol. 54, no. 9, pp. 4155–4168, Sept 2008.

[100] R. R. Tenney and N. R. Sandell, Jr., "Detection with distributed sensors," *IEEE Trans. Aerosp. Electron. Syst.*, vol. AES-17, no. 4, pp. 501–510, Jul. 1981.

[101] J. Tropp, I. Dhillon, R. Heath, and T. Strohmer, "Designing structured tight frames via an alternating projection method," *Information Theory, IEEE Transactions on*, vol. 51, no. 1, pp. 188–209, Jan 2005.

[102] J. Tsitsiklis and M. Athans, "On the complexity of decentralized decision making and detection problems," *Automatic Control, IEEE Transactions on*, vol. 30, no. 5, pp. 440–446, May 1985.

[103] J. N. Tsitsiklis, "Decentralized detection by a large number of sensors*," *Math. control, Signals, and Systems*, vol. 1, pp. 167–182, 1988.

[104] P. K. Varshney, *Distributed Detection and Data Fusion*. New York:Springer-Verlag, 1997.

[105] V. V. Veeravalli and P. K. Varshney, "Distributed inference in wireless sensor networks," *Phil. Trans. R. Soc. A*, vol. 370, no. 1958, pp. 100–117, Jan. 2012.

[106] A. Vempaty, K. Agrawal, H. Chen, and P. K. Varshney, "Adaptive learning of byzantines' behavior in cooperative spectrum sensing," in *Proc. IEEE Wireless Comm. and Networking Conf. (WCNC)*, march 2011, pp. 1310 –1315.

[107] A. Vempaty, Y. Han, and P. Varshney, "Target Localization in Wireless Sensor Networks Using Error Correcting Codes," *IEEE Trans Inf. Theory,*, vol. 60, no. 1, pp. 697–712, Jan 2014.

[108] A. Vempaty, L. Tong, and P. K. Varshney, "Distributed inference with Byzantine data: state-of-the-art review on data falsification attacks," *IEEE Signal Process. Mag.*, vol. 30, no. 5, pp. 65–75, Sep. 2013.

[109] A. Vempaty, B. Chen, and P. K. Varshney, "Optimal quantizers for distributed Bayesian estimation," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process. (ICASSP 2013)*, May 2013, pp. 4893–4897.

[110] A. Vempaty, H. He, B. Chen, and P. K. Varshney, "On quantizer design for distributed Bayesian estimation in sensor networks," *IEEE Trans. Signal Process.*, vol. 62, no. 20, pp. 5359–5369, Oct. 15 2014.

[111] A. Vempaty, P. Ray, and P. K. Varshney, "False discovery rate based distributed detection in the presence of Byzantines," *IEEE Trans. Aerosp. Electron. Syst.*, vol. 50, no. 3, pp. 1826–1840, Jul. 2014.

[112] A. Vempaty, L. Tong, and P. K. Varshney, "Distributed inference with Byzantine data: State-of-the-art review on data falsification attacks," *IEEE Signal Process. Mag.*, vol. 30, no. 5, pp. 65–75, Sep. 2013.

[113] P. Venkitasubramaniam, L. Tong, and A. Swami, "Quantization for maximin ARE in distributed estimation," *IEEE Trans. Signal Process.*, vol. 55, no. 7, pp. 3596–3605, Jul. 2007.

[114] J. Vila-Forcen, A. Artes-Rodriguez, and J. Garcia-Frias, "Compressive sensing detection of stochastic signals," in *42nd Annual Conference on Information Sciences and Systems (CISS).*, March 2008, pp. 956–960.

[115] R. Viswanathan and P. K. Varshney, "Distributed detection with multiple sensors: Part I – Fundamentals," *Proc. IEEE*, vol. 85, no. 1, pp. 54–63, Jan. 1997.

[116] T.-Y. Wang, L.-Y. Chang, D.-R. Duh, and J.-Y. Wu, "Distributed fault-tolerant detection via sensor fault detection in sensor networks," in *Information Fusion, 2007 10th International Conference on*, July 2007, pp. 1–6.

[117] T.-Y. Wang, Y. S. Han, P. K. Varshney, and P.-N. Chen, "Distributed Fault-Tolerant Classification in Wireless Sensor Networks," *IEEE J Sel. Areas Comm.*, vol. 23, no. 4, pp. 724 – 734, April 2005.

[118] Y.-G. Wang, Z. Liu, L. Yang, and W.-L. Jiang, "Generalized compressive detection of stochastic signals using neymanpearson theorem," *Signal, Image and Video Processing*, pp. 1–10, 2014.

[119] T. Wimalajeewa, H. Chen, and P. Varshney, "Performance analysis of stochastic signal detection with compressive measurements," in *Signals, Systems and Computers (ASILOMAR), 2010 Conference Record of the Forty Fourth Asilomar Conference on*, Nov 2010, pp. 813–817.

[120] A. D. Wyner, "The wire-tap channel," *Bell System Technical Journal*, vol. 54, no. 8, pp. 1355–1387, 1975. [Online]. Available: http://dx.doi.org/10.1002/j.1538-7305.1975.tb02040.x

[121] Q. Yan, M. Li, T. Jiang, W. Lou, and Y. Hou, "Vulnerability and protection for distributed consensus-based spectrum sensing in cognitive radio networks," in *INFOCOM, 2012 Proceedings IEEE*, March 2012, pp. 900–908.

[122] F. Yu, M. Huang, and H. Tang, "Biologically inspired consensus-based spectrum sensing in mobile Ad Hoc networks with cognitive radios," *Network, IEEE*, vol. 24, no. 3, pp. 26–30, May 2010.

[123] F. Yu, H. Tang, M. Huang, Z. Li, and P. Mason, "Defense against spectrum sensing data falsification attacks in mobile ad hoc networks with cognitive radios," in *Military Communications Conference, 2009. MILCOM 2009. IEEE*, Oct 2009, pp. 1–7.

[124] R. Zahedi, A. Pezeshki, and E. K. Chong, "Measurement design for detecting sparse signals," *Physical Communication*, vol. 5, no. 2, pp. 64 – 75, 2012, compressive Sensing in Communications. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S1874490711000541

[125] Q. Zhang, P. Varshney, and R. Wesel, "Optimal bi-level quantization of i.i.d. sensor observations for binary hypothesis testing," *Information Theory, IEEE Transactions on*, vol. 48, no. 7, pp. 2105 –2111, Jul 2002.

[126] Q. Zhang and P. K. Varshney, "Decentralized M-ary detection via hierarchical binary decision fusion ," *Information Fusion*, vol. 2, no. 1, pp. 3 – 16, Mar. 2001.

[127] W. Zhang, Z. Wang, Y. Guo, H. Liu, Y. Chen, and J. Mitola, "Distributed Cooperative Spectrum Sensing Based on Weighted Average Consensus," in *Global Telecommunications Conference (GLOBECOM 2011), 2011 IEEE*, Dec 2011, pp. 1–6.

[128] Z. Zhang, E. Chong, A. Pezeshki, W. Moran, and S. Howard, "Detection performance in balanced binary relay trees with node and link failures," *IEEE Trans. Signal Process.*, vol. 61, no. 9, pp. 2165–2177, May 2013.

[129] X. Zhu, Y. Yuan, C. Rorres, and M. Kam, "Distributed M-ary hypothesis testing with binary local decisions," *Information Fusion*, vol. 5, no. 3, pp. 157 – 167, 2004.

VITA

NAME OF AUTHOR: Bhavya Kailkhura

PLACE OF BIRTH: Karnaprayag, Uttarakhand, India

DATE OF BIRTH: June 19, 1990

GRADUATE AND UNDERGRADUATE SCHOOLS ATTENDED:

Syracuse University, NY, USA

Nagpur University, India

DEGREES AWARDED: M. S., 2012, Syracuse University, NY, USA

B. E., 2010, Nagpur University, India