Syracuse University

SURFACE

Dissertations - ALL

SURFACE

January 2015

Reliable Inference from Unreliable Agents

Aditya Vempaty
Syracuse University

Follow this and additional works at: https://surface.syr.edu/etd

Part of the Engineering Commons

Recommended Citation

Vempaty, Aditya, "Reliable Inference from Unreliable Agents" (2015). *Dissertations - ALL*. 332.
https://surface.syr.edu/etd/332

This Dissertation is brought to you for free and open access by the SURFACE at SURFACE. It has been accepted for inclusion in Dissertations - ALL by an authorized administrator of SURFACE. For more information, please contact surface@syr.edu.

ABSTRACT

Distributed inference using multiple sensors has been an active area of research since the emergence of wireless sensor networks (WSNs). Several researchers have addressed the design issues to ensure optimal inference performance in such networks. The central goal of this thesis is to analyze distributed inference systems with potentially unreliable components and design strategies to ensure reliable inference in such systems. The inference process can be that of detection or estimation or classification, and the components/agents in the system can be sensors and/or humans. The system components can be unreliable due to a variety of reasons: faulty sensors, security attacks causing sensors to send falsified information, or unskilled human workers sending imperfect information. This thesis first quantifies the effect of such unreliable agents on the inference performance of the network and then designs schemes that ensure a reliable overall inference.

In the first part of this thesis, we study the case when only sensors are present in the system, referred to as sensor networks. For sensor networks, the presence of malicious sensors, referred to as Byzantines, are considered. Byzantines are sensors that inject false information into the system. In such systems, the effect of Byzantines on the overall inference performance is characterized in terms of the optimal attack strategies. Game-theoretic formulations are explored to analyze two-player interactions.

Next, Byzantine mitigation schemes are designed that address the problem from the system's perspective. These mitigation schemes 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. When such schemes are not possible, Byzantine tolerant schemes using error-correcting codes are developed that tolerate the effect of Byzantines and maintain good performance in the network. Error-correcting codes help in correcting the erroneous information from these Byzantines and thereby counter their attack.

The second line of research in this thesis considers humans-only networks, referred to as human networks. A similar research strategy is adopted for human networks where, the effect of unskilled humans sharing beliefs with a central observer called *CEO* is analyzed, and the loss in performance due to the presence of such unskilled humans is characterized. This problem falls under the family of problems in information theory literature referred to as the *CEO Problem*, but for belief sharing. The asymptotic behavior of the minimum achievable mean squared error distortion at the CEO is studied in the limit when the number of agents L and the sum rate R tend to infinity. An intermediate regime of performance between the exponential behavior in discrete CEO problems and the $1/R$ behavior in Gaussian CEO problems is established. This result can be summarized as the fact that sharing beliefs (uniform) is fundamentally easier in terms of convergence rate than sharing measurements (Gaussian), but sharing decisions is even easier (discrete).

Besides theoretical analysis, experimental results are reported for experiments designed in collaboration with cognitive psychologists to understand the behavior of humans in the network. The act of fusing decisions from multiple agents is observed for humans and the behavior is statistically modeled using hierarchical Bayesian models. The implications of such modeling on the design of large human-machine systems is discussed. Furthermore, an error-correcting codes based scheme is proposed to improve system performance in the presence of unreliable humans in the inference process. For a crowdsourcing system consisting of unskilled human workers providing unreliable responses, the scheme helps in designing easy-to-perform tasks and also mitigates the effect of erroneous data. The benefits of using the proposed approach in comparison to the majority voting based approach are highlighted using simulated and real datasets.

In the final part of the thesis, a human-machine inference framework is developed where humans and machines interact to perform complex tasks in a faster and more efficient manner. A mathematical framework is built to understand the benefits of human-machine collaboration. Such a study is extremely important for current scenarios where humans and machines are constantly interacting with each other to perform even the simplest of tasks. While machines perform best in some tasks, humans still give better results in tasks such as identifying new patterns. By using humans and machines together, one can extract complete information about a phenomenon of interest. Such an architecture, referred to as Human-Machine Inference Networks (HuMaINs), provides promising results for the two cases of human-machine collaboration: *machine as a coach* and *machine as a colleague*. For simple systems, we demonstrate tangible performance gains by such a collaboration which provides design modules for larger, and more complex human-machine systems. However, the details of such larger systems needs to be further explored.

To my family:

past, present, and future.

RELIABLE INFERENCE FROM UNRELIABLE AGENTS

By

Aditya Vempaty
B.Tech., Indian Institute of Technology, Kanpur, India, 2011

DISSERTATION

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

Syracuse University
June 2015

Copyright © 2015 Aditya Vempaty

All rights reserved

ACKNOWLEDGMENTS

First and foremost, I would like to thank my advisor Prof. Pramod K. Varshney for his invaluable guidance throughout this dissertation. Since my first interaction with him as a summer intern in 2010, I have always been comfortable speaking my mind around him. His patience and belief has pushed me to work hard and encouraged me to pursue good research. Ever since he accepted me into his research group, he has been a great mentor with his continuous support and encouragement. I am also grateful to my co-advisor Prof. Lav R. Varshney for giving me the opportunity to learn from his deep insights and in-depth knowledge. I have learnt a lot from him over the past few years. I have thoroughly enjoyed discussing with him about research, community, philosophy, and future. I feel truly fortunate to have worked under his guidance during my graduate education. I am honored to be his first PhD advisee. I have had the great opportunity to discuss any part of my research with them in great detail and they always made time for any questions I had. I would not have enjoyed my graduate studies so much without their guidance, and would like to thank them.

I would like to extend my heartfelt thanks to Dr. Onur Ozdemir, Prof. Priyadip Ray, and Prof. Yunghsiang Han, for being excellent mentors to me during my PhD. I have learnt quite a lot from them during my graduate studies, especially the initial stages, which shaped me into who I am now. The ideas discussed in the thesis have been greatly influenced by them. I have sent them badly written drafts of our research, which they have patiently corrected, and have shared with me their insights. It has been a delightful journey - working, discussing, thinking, and meeting with them, over these years.

My PhD research has been supported by numerous grants from AFOSR, ARO, ARL, and Syracuse University Graduate Fellowship, for which I am grateful. In addition, I would like to thank my defense committee members Prof. Biao Chen, Prof. Yingbin Liang, Prof. Amy Criss, and Prof. Mustafa Gursoy. I have had the pleasure of working with Prof. Biao Chen during my graduate studies, and I am grateful for his wonderful guidance. Also, special thanks to Prof. Amy Criss and Dr. Gregory Koop from the Department of Psychology, without whom the results in Chapter 6 would not have been possible.

I have had several technical and philosophical discussions about statistical signal processing and research in general with Arun Subramanian. A big thanks to him for guiding me when I was a novice, and when I did not know where I am heading towards in my research. I have enjoyed the company and support of my lab mates Swarnendu, Satish, Sid, Yujiao, Raghed, Hao, Sijia, Nianxia, Bhavya, Prashant, Shan, Swatantra, Qunwei, Swastik, Sora, and Thakshila. They have listened to my boring presentations with keen interest, and have always been a constant source of inspiration to carry out my research, for which I am grateful to them. They made sure that time spent here would remain as one of the best times of my life. Our fun times would be missed!

Weekend hangouts with Gogi, BK, PK, Shommo, Deepa, Kapil, Mukund, Niteesha, Manogna, and Chitti, were lots of fun and will be greatly missed. Special thanks to Mrs. Anju Varshney who made me feel at home during my stay in Syracuse.

In the process of obtaining higher education, I have deprived my family an opportunity to spend time with me. I too have missed their company sorely over all these years. Their unconditional and extraordinary support, care, selfless sacrifice, and encouragement during this time, which they have bestowed at a great personal cost to them, cannot be emulated. I am truly and deeply thankful to my parents Anil and Radha, my sister Athni, and my brother-in-law Mallik for the same.

And last, but far from least, I would like to thank Swetha who has been the one constant in this roller-coaster ride called graduate studies. Her cheerful attitude towards life has been a great mental support for me during this time. Her support, trust, and understanding are greatly appreciated. I hope to have her by my side for the rest of my life.

TABLE OF CONTENTS

Acknowledgments **vii**

List of Tables **xvi**

List of Figures **xvii**

1 Introduction **1**

1.1 Inference from Multiple Agents 1

1.2 Research Challenges 4

1.3 Research Methodology 5

1.4 Outline and Contributions 7

2 Background **15**

2.1 General Architecture 15

2.2 Taxonomy 16

2.2.1 Inference 16

2.2.2 Agents 17

2.2.3 Unreliability 18

2.3 Past Work 19

2.4 Background Material 21

2.4.1 Distributed Classification Fusion using Error Correcting Codes 21

2.4.2 Distributed Classification using Soft-decision Decoding 22

2.5 Summary of Contributions 24

3 Estimation in Sensor Networks: Unreliable Agents 25

3.1 Introduction 25

3.2 Preliminaries 26

3.2.1 System Model 26

3.2.2 Monte Carlo Method based Target Localization 28

3.2.3 Performance Metrics 29

3.3 Localization in the Presence of Byzantine Sensors 30

3.3.1 Independent Attack Model 30

3.3.2 Blinding the Fusion Center 31

3.3.3 Best Honest and Byzantine Strategies: A Zero-Sum Game 33

3.3.4 Numerical Results 35

3.4 Collaborative Attack 36

3.5 Target Tracking in the Presence of Byzantine Sensors 41

3.5.1 System Model 41

3.5.2 Performance Metrics 43

3.5.3 Attack Model 45

3.5.4 Blinding the Fusion Center 47

3.6 Discussion 47

4 Estimation in Sensor Networks: Reliable Inference 49

4.1 Introduction 49

4.2 Byzantine Identification 50

4.2.1 Numerical Results 52

4.2.2 Discussion 53

4.3 Design of Dynamic Non-Identical Quantizers 56

4.3.1 Honest Sensors Only 56

4.3.2 Game Between Honest and Byzantine Sensors 59

4.3.3 Numerical Results 61

4.3.4 Target Tracking . . . 63

4.4 Localization as Hierarchical Classification . . . 68

4.4.1 Basic Coding Based Scheme . . . 69

4.4.2 Performance Analysis . . . 71

4.4.3 Numerical Results . . . 77

4.5 Localization in the Presence of Byzantines . . . 78

4.5.1 Exclusion Method with Weighted Average . . . 79

4.5.2 Performance Analysis . . . 80

4.5.3 Simulation Results . . . 82

4.6 Soft-Decision Decoding for Non-Ideal Channels . . . 85

4.6.1 Decoding Rule . . . 85

4.6.2 Performance Analysis . . . 86

4.6.3 Numerical Results . . . 88

4.7 Discussion . . . 91

5 Estimation in Human Networks: Unreliable Local Agents 93

5.1 Introduction . . . 93

5.2 Non-Regular CEO Problem . . . 95

5.2.1 Examples of Observation Channels Satisfying Property 5.2.1 . . . 98

5.3 Direct Coding Theorem . . . 101

5.3.1 Quantization of Alphabets . . . 102

5.3.2 Codes That Approximate the Test Channel . . . 103

5.3.3 Encoding and Decoding . . . 104

5.3.4 Further Analysis of Code Rate . . . 105

5.3.5 Estimation Scheme . . . 107

5.4 Converse Coding Theorem . . . 112

5.5 Discussion . . . 117

6 **Detection in Human Networks: Unreliable Global Fusion** **119**
6.1 Introduction . . . 119
6.2 Experiments and Data Analysis . . . 120
6.2.1 Experimental Design . . . 120
6.2.2 Data Analysis . . . 121
6.3 Bayesian Hierarchical Model . . . 123
6.3.1 Description of Model . . . 123
6.4 Optimal Design of Sociotechnical Systems . . . 124
6.5 Discussion . . . 131

7 **Classification in Human Networks: Reliable Inference** **132**
7.1 Introduction . . . 132
7.2 Coding for Crowdsourcing . . . 134
7.2.1 Reliable Classification using Crowds . . . 134
7.2.2 Unreliable Workers . . . 136
7.3 Crowdsourcing System with Individual Crowd Workers . . . 137
7.3.1 Model . . . 137
7.3.2 Performance Characterization . . . 137
7.3.3 Majority Voting . . . 140
7.3.4 Performance Evaluation . . . 142
7.4 Crowdsourcing System with Peer-dependent Reward Scheme . . . 149
7.4.1 Model . . . 149
7.4.2 Performance Characterization . . . 149
7.4.3 Majority Voting . . . 151
7.4.4 Performance Evaluation . . . 154
7.5 Crowdsourcing System with Common Sources of Information . . . 155
7.5.1 Model . . . 156
7.5.2 Performance Characterization . . . 157

7.5.3 Performance Evaluation 160

7.6 Conclusion 162

8 Human-Machine Inference Networks (HuMaINs) 164

8.1 Introduction 164

8.2 Knowledge Discovery Problem 166

8.2.1 Problem Formulation 167

8.2.2 Size of Knowledge Base 168

8.2.3 Quality of Knowledge Base 175

8.2.4 Empirical Observations 178

8.3 Solution Search Problem 180

8.3.1 Mathematical Formulation 180

8.3.2 Collaborative Problem Solving 181

8.3.3 Analysis 182

8.4 Discussion 184

9 Conclusion 185

9.1 Summary 185

9.2 Future Directions 188

9.2.1 Statistical Modeling of Human Decision Making 192

9.2.2 Design of Robust Algorithms for Collaborative Decision Making 192

A Appendix 194

A.1 Proof of Proposition 4.3.1 194

A.2 Proof of Proposition 4.3.3 195

A.3 Proof of Lemma 4.4.2 196

A.4 Proof of Theorem 4.4.4 197

A.5 Proof of Theorem 4.6.3 199

A.6 Proposition A.6.1 203

A.7 Proposition A.7.1 205

References **206**

LIST OF TABLES

2.1 Summary of contributions 24

4.1 Performance of the Byzantine identification scheme 54
4.2 System design based on the required performance 82

7.1 Performance of proposed approach using real data 148

LIST OF FIGURES

1.1 General system model of distributed inference 2

1.2 Research methodology for reliable inference from unreliable agents 6

2.1 Inference with potentially unreliable agents 16

3.1 System model for target localization in wireless sensor networks 27

3.2 Trace of inverse of Fisher information of target location with fraction of Byzantines 36

3.3 Determinant of Fisher information of target location with fraction of Byzantines . . 37

3.4 Mean square error of target location's estimate with fraction of Byzantines 37

3.5 Surface plot of trace of inverse of Fisher information of target location 38

3.6 Contour plot of trace of inverse of Fisher information of target location 39

3.7 Surface plot of determinant of Fisher information of target location 40

3.8 Contour plot of determinant of Fisher information of target location 41

4.1 Number of wrongly identified sensors with time 54

4.2 Estimate of fraction of Byzantines in the network 55

4.3 Region of Interest showing the sensors that cannot be perfectly identified using the learning algorithm 55

4.4 MSE comparison of the two schemes: identical threshold scheme and non-identical dynamic threshold scheme 62

4.5 Target tracking with 10% Byzantines in the network. 66

4.6 Target tracking with 30% Byzantines in the network. 67

4.7 Tracking comparison of the two schemes in the presence of Byzantines. 68

4.8 Equal region splitting of the ROI for localization as a 4-hypothesis test 69
4.9 ROI with an example set of paired sensors 76
4.10 Region detection probability versus the standard deviation of noise with varying number of sensors 78
4.11 MSE comparison of the three localization schemes: Basic coding scheme, Coding with exclusion, and ML-based localization 83
4.12 Probability of detection of the target region with fraction of Byzantines in the network 84
4.13 MSE of the target location estimate with varying number of sensors 84
4.14 Probability of detection of target region with varying number of sensors 85
4.15 MSE comparison of the basic coding scheme using soft- and hard- decision decoding 89
4.16 Probability of detection of target region comparison of the basic coding scheme using soft- and hard- decision decoding 89
4.17 MSE comparison of the exclusion coding scheme using soft- and hard- decision decoding 90
4.18 Probability of detection of target region comparison of the exclusion coding scheme using soft- and hard- decision decoding 90
4.19 Probability of detection of target region of the exclusive coding scheme using soft-decision decoding with varying number of sensors 91

5.1 An achievability scheme for non-regular CEO problem 101

6.1 Distribution of subjects' match value between the human decision and the CV rule's decision 122
6.2 Individual-level decision model of decision fusion by people 123
6.3 Distribution of match value with different number of sources 125
6.4 Bayesian hierarchical model of decision fusion by humans 126
6.5 Hierarchical system consisting of human decision fusion components 127

6.6 Percentage improvement in system performance by using the Bayesian hierarchical model of human decision fusion 130
6.7 Percentage improvement in system performance by using the Bayesian hierarchical model of human decision fusion 131

7.1 A schematic diagram showing binary questions posed to workers and the decoding rule used by the task manager 136
7.2 Coding-based crowdsourcing system misclassification probability as a function of worker reliability 144
7.3 Misclassification probability using coding- and majority-based approaches with the spammer-hammer model for a small system 144
7.4 Misclassification probability using coding- and majority-based approaches with the spammer-hammer model for a large system 146
7.5 Misclassification probability using coding- and majority-based approaches with the Beta model of crowd for a small system 146
7.6 Misclassification probability using coding- and majority-based approaches with the Beta model of crowd for a large system 147
7.7 Misclassification probability as a function of crowd quality using optimal code matrix, random code matrix for coding-based approach and majority approach with the spammer-hammer model 147
7.8 Effect of peer-dependent reward scheme on misclassification probability 155
7.9 Misclassification probability of a crowdsourcing system with workers having common sources of information 161
7.10 Misclassification probability of a crowdsourcing system with peer-dependent reward scheme and workers having common sources of information 162

8.1 Comparison of analytical expressions and simulations of size of knowledge base . . 173
8.2 Effect of initial set of known elements on the discovery rate. 173

8.3 Effect of noise on the discovery process 174
8.4 Effect of noise on the discovery process when more probable elements are already known 175
8.5 Effect of different prior distribution on the discovery process 175
8.6 Expected fraction of element set with iterations for different total number of elements 176
8.7 Expected fraction of element set with iterations for different number of known elements 176
8.8 Expected quality of discovered elements with time 179

9.1 Elements of future work 191

CHAPTER 1

INTRODUCTION

1.1 Inference from Multiple Agents

Inferring about an unknown phenomenon based on observations of multiple agents is a part of our daily lives. Almost every decision we make is made after collecting evidence from multiple sources (agents). These sources are typically of two kinds: sensors/machines that are objective in nature and humans that are typically subjective in nature (sometimes referred to as *hard* and *soft* sources, respectively). Decision fusion is the process of integrating decisions made by multiple entities about the same phenomenon into a single final decision. The typical framework of parallel decision fusion is shown in Fig. 1.1, where a set of local decision makers (LDMs) observe a phenomenon and make decisions regarding its presence or absence (Yes/No binary decisions). These local decisions are received by a global decision maker (GDM) who fuses the received data to make the final decision.

When all the sources/agents are sensors, we have the well-studied sensor networks [20, 154, 155, 163]. Due to the advancements in wireless technology, wireless sensor networks (WSNs) are increasingly being used both in military and civilian applications [2]. One such application is to monitor, detect, and/or estimate the location of a target or object in an area of interest [101, 155, 167]. Localization techniques proposed in the literature for sensor networks include

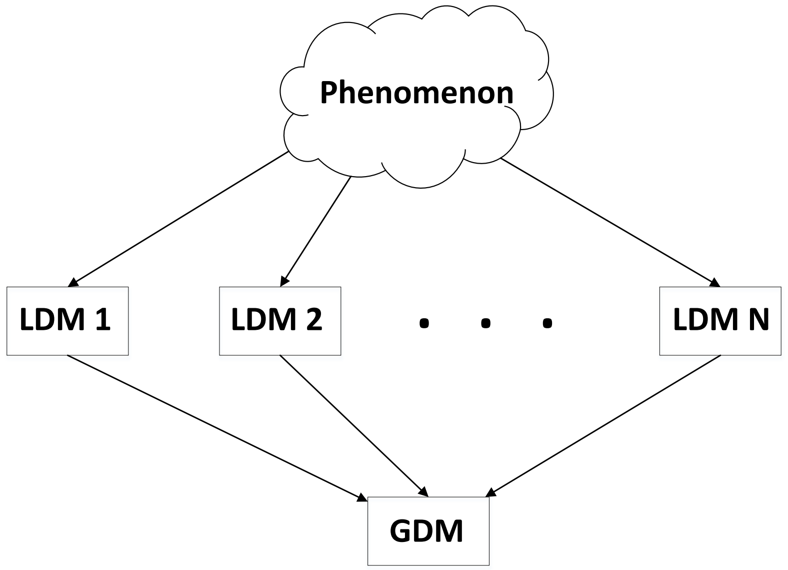


Fig. 1.1: System model consisting of local decision makers (LDMs) and a global decision maker (GDM).

direction of arrival (DOA), time of arrival (TOA) and time-difference of arrival (TDOA) based methods [68] [34]. Recent research has focused on developing techniques which do not suffer from imperfect time synchronization. For example, in [91] [92], the authors propose localization in WSNs by “dumb sensors” which are cheap and do not require time synchronization or extensive local processing. Received signal strength (RSS) based methods, which do not suffer from imperfect synchronization and/or extensive processing, have also been proposed which employ least-squares or maximum-likelihood (ML) based source localization techniques [85] [126]. Due to resource constraints such as energy and bandwidth, it is often desirable that sensors only send binary or multi-bit quantized data to limit the communication requirements. RSS based target localization using quantized data in a sensor network has been investigated in the literature [106] [101].

When all the agents in the network are humans, we have the human networks such as team decision making systems typically seen in large organizations such as firms. Consider the problem faced by the chief executive officer (CEO) of a firm with a large portfolio of projects that each have an underlying probability of success, say drawn from a uniform distribution on $[0, 1]$. Each of the subordinates will have noisy beliefs about the risks facing the projects: random variables denoting the success probability jointly distributed, e.g., according to a copula model (common in mathematical finance to model beliefs about risks [38, 53, 99]). Subordinates must convey these risks, but are not allowed to convene. The CEO has cognitive constraints that limit the information rate he/she can receive from the subordinates, requiring subordinates to partition risks into quantal grades like A, B, C, and D, before conveying them. Such quantized grading is typical in businesses with complex information technology projects [115]. Upon receiving information from the subordinate agents, the CEO estimates the underlying success probability to minimize mean squared error (Brier score [110]) in estimating risk before taking action.

Additional questions arise when the global fusion center is also a human who is fusing data from multiple humans, like the CEO described before. In such cases, it is of interest to understand the act of decision fusion by the human. It is interesting to compare the behavior of humans with optimal decision rules that have been developed based on statistical models. According to the bounded rationality [54] argument about humans, it is expected that humans do not necessarily follow optimal rules. By asking how far from optimality humans are at fusing data from multiple sources, we can develop bounded rational models for this task.

The problem of fusing multiple human decisions has been investigated in different contexts in the psychology literature (see [131, 132] and references therein). Such a framework is very similar to the problems of social choice theory, voting, etc. Based on the observations made from these works, it can be expected that human behavior is non-deterministic in general. Therefore, Bayesian modeling approaches [56] can be adopted to model such behavior.

When only the global decision maker is a machine and the local decision makers are humans, the above framework addresses the paradigm of crowdsourcing for distributed inference tasks [22, 136, 137] where multiple crowd workers provide their responses regarding a task of interest. Conventional studies of human decision making involve decisions made by individuals such as military commanders or company presidents, or by a small number of individuals such as committees or cabinets operating under decision fusion rules such as majority voting. Behavioral scientists have studied human decision making from a cognitive viewpoint whereas economists have studied it from a decision theory viewpoint. When agents are rational and knowledgeable about the problem domain, team decision theory provides well-developed insights [94]. With the emergence of widespread and inexpensive computing and networking infrastructure, new paradigms for human participation, such as crowdsourcing, have arisen for distributed inference tasks [22, 136, 137] and are becoming very prevalent in the specialized and globalized knowledge economy.

1.2 Research Challenges

While there are several practical challenges one faces while inferring based on observations/information from multiple agents, this thesis focuses on the class of challenges associated with the lack of reliable information. We focus on the causes of unreliable data from the agents. These causes can be divided into three types: faulty or malicious sensors (Byzantines), imperfect channels in the network, and unskilled humans.

Secure inference using sensors is extremely important in situations where malicious sensors attempt to disrupt the network and diminish its capability. Several algorithms have been developed for secure localization in WSNs [24,86,173]. While several algorithms have also been designed for localization using quantized measurements [101], no malicious sensors or attackers in the network have been considered. One kind of attack in a WSN is by Byzantines [159] where an adversary takes over some sensors of the WSN and forces them to send falsified information to the fusion center (FC). The main goal of Byzantine attackers is to undermine the network such that the FC is unable to estimate the correct location of the target. Although the presence of such malicious sensors is the focus of this thesis, there are other related security challenges such as jamming and eavesdropping that are relevant to the problems considered in this thesis but are not considered here.

An important element of WSNs is the presence of non-ideal wireless channels between sensors and the FC [30] [108]. These non-ideal channels corrupt the quantized data sent by the local sensors to the FC. This causes errors which deteriorates the inference performance at the FC. Hence, the inference system must be supported by robust coding and modulation schemes that efficiently model these vastly different channel characteristics. Also, in most of the cases, the optimal system parameter values change due to the presence of imperfect channels in the system. For example, when a single-sensor is performing an M-ary ($M > 2$) hypothesis testing problem, or when multiple sensors are performing a hypothesis testing problem, optimal sensor signaling is dependent on the channels in the system [30]. Therefore, a system designed in the absence of imperfect channels can have a degraded performance in the presence of non-ideal channels. In particular, in a typical WSN where two or more sensors are engaged in the detection problem, channel-aware design always leads to performance improvement under given resource constraints [87]. Thus, the presence of imperfect channels in WSNs presents an important research challenge to the problems considered in this thesis.

Although both crowdsourcing and conventional team decision making [94] involve human decision makers, there are three major differences. First, the number of participants involved in crowdsourcing is usually large. Second, contrary to traditional mathematical models of team decision making [154], members of the crowd may be unreliable or malicious [62, 84, 152] especially since they are often anonymous [47, 122]. Third, workers may not have sufficient domain expertise to perform full classification and may only be able to make simpler binary distinctions [25]. These differences give rise to a multitude of new system design challenges when crowdsourcing is employed for tasks with varying quality requirements and time deadlines. The main challenge, however, is *quality control* to ensure reliable crowd work [62] with provable performance guarantees. Also of importance are the privacy issues of humans taking part in the task and the tasks themselves. Although the major focus of this thesis is on the case when the humans are unskilled, some relevant work on task privacy can be found in [152, 153].

1.3 Research Methodology

The problem of attaining reliable performance from unreliable components is not a new one. Claude Shannon in his unpublished manuscript of 1956 titled “Reliable Machines from Unreliable Components" considers the problem of designing reliable machines from unreliable components [125]. He very rightly suggests that there are typically three methods to improve system reliability: complete redesign, improve system components, and/or use of error-correction codes (Fig. 1.2). Complete redesign refers to changing the design of the system as a whole, for example moving from analog systems to digital systems. System components can be improved by using more robust machines. And finally, reliable machines can be designed by using error-correcting codes. Some other researchers have also addressed similar problems of attaining reliable performance from unreliable data [140, 150]. In [140], Taylor addresses the problem of theoretical capabilities of memories and computing systems that are designed from unreliable components. More recently, communication systems with unreliable decoders has been considered in [150] where the performance of such communication systems has been studied.

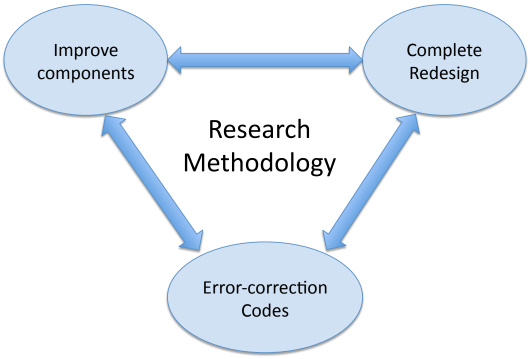


Fig. 1.2: Research methodology for reliable inference from unreliable agents

However, to the best of the author's knowledge, researchers have not considered this approach for distributed inference networks consisting of humans and/or machines. In such systems, one can use the methodology described above (Fig. 1.2) to attain reliable performance. As seen later in the thesis, complete redesign might correspond to a total change in the inference architecture or process. System components can be improved either by identifying the malicious sensors and removing them or improving the performance of humans by giving them easier tasks. Although the cause of unreliable information is different for humans and machines, the effect is the same: errors in the data. Therefore, one can use coding-theory ideas to correct these errors and improve performance. The schemes proposed show the benefit of adapting coding based techniques to signal processing applications.

1.4 Outline and Contributions

The central goal of this thesis is to analyze the performance of inference systems with potentially unreliable components and design strategies to ensure reliable inference performance from these systems with unreliable agents. Since the inference process can be one of detection or estimation or classification, it brings in several problems of interest. An overview of the general architecture of problems is provided in Chapter 2 and literature review and background material needed for the later chapters of the thesis is presented. Sensor networks are studied in Chapters 3 and 4 while human networks are considered in Chapters 5, 6, and 7 . For each of these parts (Chapters 3-4 and Chapters 5-7), we follow the research methodology described in Sec. 1.3. In Chapter 3, the effect of unreliable components of the sensor network is analyzed to determine their effect on the overall performance of the system. This study leads to Chapter 4 where a redesign of the system is considered and learning-based schemes along with error-correcting codes are used to counter the unreliable data from malicious sensors. A similar strategy is followed for human networks where, in Chapters 5 and 6, the effect of humans in the system is analyzed and quantified. In Chapter 7, we propose ways to improve the system components by redesigning the system where humans perform easy tasks and error-correcting codes are used to mitigate the effect of potential unreliable data. The human-machine inference framework is developed in Chapter 8 where humans and machines interact together to perform complex tasks in a faster and more efficient manner. 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 explained in this chapter.

Chapter 3: Estimation in Sensor Networks: Unreliable Agents

In this chapter, we consider the case of sensor networks performing a location estimation task in WSNs using binary quantized data. We consider malicious sensors called Byzantines and investigate the target localization problem under a Bayesian framework. We assume the target location to be random and develop a Monte Carlo based approach for target localization. The appropriate performance metric to characterize the performance of the network is Posterior Fisher Information or Posterior Cramér-Rao lower bound (PCRLB). Two kinds of attack strategies are considered: Independent and Collaborative attacks. By modeling the effect of Byzantines as a binary symmetric channel (BSC), we determine the fraction of Byzantine attackers in the network that make the FC incapable of utilizing sensor information to estimate the target location. Optimal attacking strategies for given attacking resources are also derived by modeling the behavior as a zero-sum game. We use PCRLB as the utility function and find the Nash Equilibrium as the saddle point. Results are also extended to the case of target tracking problem.

Chapter 4: Estimation in Sensor Networks: Reliable Inference

The design of schemes for ensuring reliable performance from systems consisting of malicious agents is considered in this chapter. Multiple schemes are proposed for the mitigation of Byzantine attacks. The first scheme is based on a Byzantine identification method under the assumption of identical local quantizers. We show with simulations that the proposed learning-based scheme identifies most of the Byzantines. In order to improve performance, we propose a second scheme in conjunction with our identification scheme where dynamic non-identical threshold quantizers are used at the sensors. We show that it not only reduces the location estimation error but also makes the Byzantines *ineffective* in their attack strategy. We also propose the use of coding theory techniques to estimate the location of the target in WSNs. We develop the fundamental theory and derive asymptotic performance results. We first consider the code design problem in the absence of channel errors and Byzantine data. The proposed scheme models the localization problem as a hierarchical classification problem. The scheme provides a coarse estimate in a computationally efficient manner as compared to the traditional ML based approach. We present performance analysis of the proposed scheme in terms of the detection probability of the correct region. We show analytically that the schemes achieve perfect performance in the asymptotic regime. We address the issues of Byzantines and channel errors subsequently and modify our scheme to handle them. The error correction capability of the coding based approach provides Byzantine tolerance capability and the use of soft-decoding at the FC provides tolerance to channel errors.

Chapter 5: Estimation in Human Networks: Unreliable Local Agents

In Chapter 5, we consider networks with humans and analyze the effect of unreliable local humans in networks. When all local agents are humans, it can be regarded as the situation in a large organization such as a firm where a group of agents independently observe corrupted versions of data and transmit coded versions over rate-limited links to a CEO. The CEO then estimates the underlying data based on the received coded observations. Agents are not allowed to convene before transmitting their observations. This falls under the category of problems referred to as the CEO problem, but with non-regular source distributions (in the sense of Bayesian estimation theory [145, p. 72]), and observations governed by a given conditional probability density function, for example, through copula models. More precisely an i.i.d. source sequence $X(t)$ is considered, which follows a probability density function (pdf) $f_X(x)$ with finite support $\mathcal{X}$, such that

$$\frac{\partial f_X(x)}{\partial x} \text{ or } \frac{\partial^2 f_X(x)}{\partial x^2}$$

either does not exist or is not absolutely integrable. The asymptotic behavior of quadratic distortion as a function of sum rate in the limit of large numbers of agents and rate is determined. As commented by Viswanathan and Berger [162], results for discrete and continuous alphabets are not very different in most problems of information theory. However, for the CEO problem, the average distortion decays at an exponential rate for the discrete case and decays as $1/R$ for the

Gaussian case, where R is the sum rate. In this chapter, we derive an intermediate $1/R^2$ decay rate behavior when the regularity conditions required for the Bayesian Cramér-Rao lower bound used in [162] do not hold.

Chapter 6: Detection in Human Networks: Unreliable Global Fusion

The effect of a human fusing decisions from multiple agents is considered in Chapter 6. The problem of decision fusion by humans is addressed and results of experiments conducted on human subjects, to understand this human behavior, are reported. Also, a hierarchical Bayesian model is developed to capture the decision fusion by people. Due to the hierarchical nature, this model encompasses the differences observed at various levels: individual level, crowd level, and population level. On an individual level, every human has a different bias which affects his/her decision fusion process. A crowd is a collection of people who have similar understanding due to cultural, societal, or other factors, and therefore, might have similar characteristics in performing tasks. On a population level, there are differences in societies, cultures, or demographics, which affect the decision fusion process. By adopting a hierarchical Bayesian model, these various types of differences can be modeled. The implications of such a model on developing large-scale human-machine systems are presented by developing optimal decision fusion trees with both human and machine agents.

Chapter 7: Classification in Human Networks: Reliable Inference

In Chapter 7, we discuss ways to enhance the performance of networks with humans by developing easy-to-answer questions. By focusing on crowdsourcing of M-ary classification tasks, such as multi-class object recognition from images into fine-grained categories [25], we design a coding-theoretic scheme to minimize misclassification probability. Distributed classification codes and a minimum Hamming distance decoder is used such that workers need to only answer binary questions. The efficacy of this coding-based approach is demonstrated using simulations and through real data from Amazon Mechanical Turk [130], a paid crowdsourcing microtask platform. The approach is analyzed under different crowdsourcing models including the peer-dependent reward scheme [61] and the dependent observations model [111]. In the process, an ordering principle for the quality of crowds is also developed. For systems with peer-dependent reward schemes, it is observed that higher correlation among workers results in performance degradation. Further, if the workers also share dependent observations due to common sources of information, it is shown that the system performance deteriorates as expected. However, it is also observed that when the observations become independent, the performance gain due to our coding-based approach over the majority-vote approach increases.

Chapter 8: Human-Machine Inference Networks (HuMaINs)

A human-machine collaborative paradigm is developed in this chapter where humans and machines, which have radically different cognitive strengths and weaknesses, are combined in an intelligent manner to tackle various informational tasks with high speed and accuracy. A general problem solving architecture is considered and possible models of collaboration are outlined. Based on this architecture of human-machine inference networks, two example problems are considered. In the knowledge discovery problem, a human interested in discovering all the unknown elements of a set is supported by a machine. This partnership is referred to as *machine as a coach* collaboration. The performance of this learning process is characterized in terms of size and quality of the knoweldge base at every time step. In the solution search problem, humans and machines collaborate as *colleagues* to determine the solution to a problem, such as finding a maximum point for a given function. The mathematical frameworks presented in this chapter provide an intuitive understanding of the benefits of human-machine collaboration and can help in the design of larger human-machine inference networks.

Chapter 9: Conclusion

In the conclusion 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.

Bibliographic Note

Parts of Chapter 3 appear in the conference papers:

- K. Agrawal, A. Vempaty, H. Chen, and P. K. Varshney, “Target localization in sensor networks with quantized data in the presence of Byzantine attacks," in *Proceedings of Asilomar Conference on Signals, Systems and Computers*, pp. 1669–1673, Nov. 2011.
- A. Vempaty, O. Ozdemir, and P. K. Varshney, “Mitigation of Byzantine attacks for target location estimation in wireless sensor networks," in *Proceedings of the 46th Annual Conference on Information Sciences and Systems (CISS)*, Mar. 2012.
- A. Vempaty, O. Ozdemir, and P. K. Varshney, “Target tracking in wireless sensor networks in the presence of Byzantines," in *Proceedings of International Conference on Information Fusion (FUSION)*, Jul. 2013.

and in the journal paper:

- A. Vempaty, O. Ozdemir, K. Agrawal, H. Chen, and P. K. Varshney, “Localization in Wireless Sensor Networks: Byzantines and Mitigation Techniques," *IEEE Transaction on Signal Processing*, vol. 61, no. 6, pp. 1495–1508, Mar. 15, 2013.

Parts of Chapter 4 appear in the conference papers:

- A. Vempaty, O. Ozdemir, and P. K. Varshney, “Mitigation of Byzantine attacks for target location estimation in wireless sensor networks," in *Proceedings of the 46th Annual Conference on Information Sciences and Systems (CISS)*, Mar. 2012.
- A. Vempaty, Y. S. Han, and P. K. Varshney, “Target Localization in Wireless Sensor Networks using Error Correcting Codes in the Presence of Byzantines," in *Proceedings of IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, pp. 5195–5199, May 2013.

- A. Vempaty, O. Ozdemir, and P. K. Varshney, “Target tracking in wireless sensor networks in the presence of Byzantines," in *Proceedings of International Conference on Information Fusion (FUSION)*, Jul. 2013.
- A. Vempaty, Y. S. Han, and P. K. Varshney, “Byzantine Tolerant Target Localization in Wireless Sensor Networks Over Non-Ideal Channels," in *Proceedings of 13th International Symposium on Communications and Information Technologies (ISCIT 2013)*, pp. 407–411, Sep. 2013.

and in the journal papers:

- A. Vempaty, O. Ozdemir, K. Agrawal, H. Chen, and P. K. Varshney, “Localization in Wireless Sensor Networks: Byzantines and Mitigation Techniques," *IEEE Transaction on Signal Processing*, vol. 61, no. 6, pp. 1495–1508, Mar. 15, 2013.
- A. Vempaty, Y. S. Han, and P. K. Varshney, “Target Localization in Wireless Sensor Networks using Error Correcting Codes," *IEEE Transactions on Information Theory*, vol. 60, no. 1, pp. 697–712, Jan. 2014.

Parts of Chapter 5 appear in the conference paper:

- A. Vempaty and L. R. Varshney, “CEO Problem for Belief Sharing," in *Proceedings of 2015 Information Theory Workshop (ITW 2015)*, Apr. 2015.

and in the journal paper:

- A. Vempaty and L. R. Varshney, “The Non-Regular CEO Problem," *IEEE Transactions on Information Theory*, vol. 61, no. 5, pp. 2764–2775, May 2015.

Parts of Chapter 6 appear in the paper:

- A. Vempaty, G. J. Koop, A. H. Criss, and P. K. Varshney, “How efficient are we at fusing decisions?" in *Conference on Digital Experimentation: CODE@MIT*, Oct. 10–11 2014.

and in the active conference manuscript:

- A. Vempaty, L. R. Varshney, G. J. Koop, A. H. Criss, and P. K. Varshney, “Decision Fusion by People: Experiments, Models, and Sociotechnical System Design", submitted to *IEEE Global Conference on Signal and Information Processing (GlobalSIP)*, Orlando, Florida, Dec. 2015.

Parts of Chapter 7 appear in the paper:

- A. Vempaty, L. R. Varshney, and P. K. Varshney, “Reliable Classification by Unreliable Crowds," in *Proceedings of IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, pp. 5558–5562, May 2013.

and in the journal paper:

- A. Vempaty, L. R. Varshney, and P. K. Varshney, “Reliable Crowdsourcing for Multi-Class Labeling using Coding Theory," *IEEE Journal of Selected Topics in Signal Processing*, vol. 8, no. 4, pp. 667–679, Aug. 2014.

CHAPTER 2

BACKGROUND

As discussed in the introduction, distributed inference has been extensively studied by various authors in the past few decades. In the context of distributed inference with multiple sensors in a sensor network, a good survey can be found in [155] and references therein. When the agents performing the inference task are humans, this setup is similar to that of team decision making studied by Radner in [113]. However, limited work has focused on the case when these agents are potentially unreliable. 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. We present some prior results for distributed inference with malicious sensors (Byzantines) present in the literature 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 can be seen in Fig. 2.1 where multiple agents observe an unknown phenomenon that they are to make inferences about. These agents can be reliable or unreliable. For example, in Fig. 2.1, there are some agents that are providing false information. Some humans can be genuinely interested in providing the right information but due to erroneous sensing (for example, using ‘eyes’ instead of ‘ears’ in the figure), the agent 2 is providing some unreliable information. On the other hand, some ‘lazy’ agents could randomly provide the information due to lack of interest (agent 3 in Fig. 2.1). Similarly, some sensors could be imperfect, or even faulty (agent 6 that is stuck-at ‘1’). Sensors are also prone to malicious attacks and could, therefore, be attacked by an external adversary and re-programmed to send flipped version of information (agent 5 in Fig. 2.1).

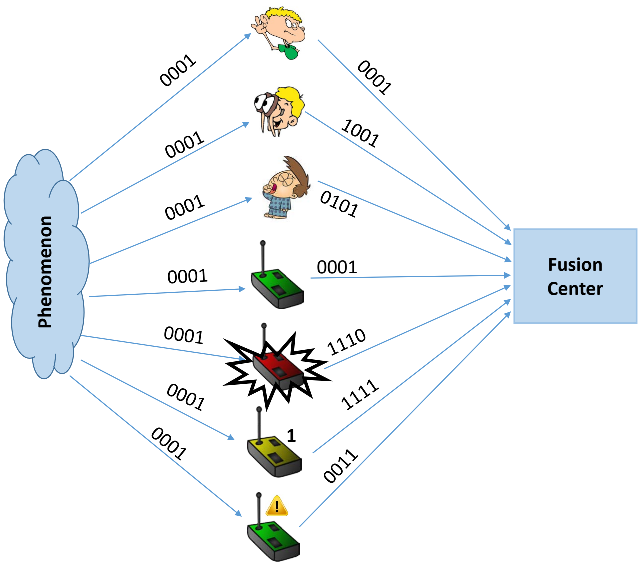


Fig. 2.1: Inference with potentially unreliable agents

2.2 Taxonomy

2.2.1 Inference

An inference problem can be of several types. The typical inference problems are of two major kinds: detection (or more generally, classification) and estimation [73, 74]. In classification, the phenomenon to be inferred is of finite possibilities, say M possible classes. This is represented by M possible hypotheses: $\mathcal{H}_0, \ldots, \mathcal{H}_{M-1}$. Multiple observations $\mathbf{x} = [x_1, \ldots, x_N]$ are collected regarding the phenomenon which follow distribution $p(\mathbf{x}|\mathcal{H}_l)$ for each hypothesis $\mathcal{H}_l$, $l = 0, \ldots, M-1$. The true class is then determined using the M-ary maximum a posteriori

(MAP) decision rule as, decide $\mathcal{H}_k$ if

$$p(\mathcal{H}_k|\mathbf{x}) > p(\mathcal{H}_i|\mathbf{x}), i \neq k.$$

For the case of equiprobable prior, this becomes the M-ary maximum likelihood (ML) decision rule. When the number of hypotheses $M = 2$, it is referred to as the detection problem.

In a typical estimation problem, the goal is to estimate the value of an unknown parameter θ (which could possibly be a vector). This value is continuous and can take any value in the set Θ. Similar to the classification problem, multiple observations are taken $\mathbf{x} = [x_1, \ldots, x_N]$ that follow a conditional distribution $p(\mathbf{x}|\theta)$ and the estimate $\hat{\theta}$ is made as the MAP estimator

$$\hat{\theta} = \arg\max_{\theta} p(\theta|\mathbf{x}).$$

When the prior is uniform, it is the ML estimator. If the parameter θ is time varying, it is typically referred to as the tracking problem.

This thesis focuses on such inference problems where the goal is to infer the unknown phenomenon using multiple observations.

2.2.2 Agents

In the case of distributed inference (classification or estimation), the observations are made in a distributed manner using multiple agents that are spatially distributed in the network. These agents observe the phenomenon and transmit their observations over (possibly) imperfect channels to FC who then fuses the data to infer about the phenomenon. The typical goal in such a framework is to design the signal processing schemes at the local sensors and the FC to infer regarding an event as accurately as possible, subject to an alphabet-size constraint on the messages transmitted by each local sensor (for a comprehensive survey, see [155] and references therein). Several aspects of such a framework can be studied [155]: 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 [20, 33, 69, 80, 89, 119, 141, 154, 157, 160, 163]. The advancement of wireless sensor networks (WSNs) renewed interest in this area along with new research challenges: wireless channels [28, 30], network topologies [3, 135, 139], sensor resource management [5, 6, 64, 114], correlated observations [27, 42, 63, 134], etc. The effect of wireless channels can be addressed by analyzing the system under the channel-aware formulation [30, 105]. Another dimension which has received vast interest is the effect of topologies on system performance. Researchers have considered the move from the traditional parallel topology (also referred to as the star topology [154]) to other topologies such as serial/tandem topologies [135] and tree topologies [139].

Depending on the problem of interest, these agents are of two types: physical sensors making objective measurements or human workers providing subjective opinions. Most of the work in the WSN literature has been when these agents are physical sensors. More recently, the crowdsourcing paradigm is considering the case when the distributed agents are human workers performing a distributed inference task. The data transmitted by the local agents to the FC is quantized due to practical constraints such as limited energy or bandwidth in the case of physical sensors and/or cognitive constraints in the case of human agents. The physical sensors are sometimes referred to as ‘hard’ sensors and the humans are sometimes referred to as ‘soft’ sensors.

This thesis focuses on distributed inference problems with observations from multiple agents of both types, namely sensors and humans.

2.2.3 Unreliability

These multiple agents in the distributed inference network are not necessarily reliable. For example, in the case of physical sensors, they could be unreliable due to noisy local observations received at the local sensors. This is governed by the conditional distribution described above. Besides this basic cause, there are several other causes for unreliable data from such physical sensors. The sensors could have permanent faults such as a stuck-at faults which causes the sensor to always send the same information to the FC irrespective of what it senses. Such a behavior of the sensor provides no information to the FC and therefore, needs to be addressed. A more malicious cause could be the case of security attacks where the sensor can be attacked and reprogrammed by an external adversary into sending false data to the FC. Such a malicious sensor sending false information would result in a deteriorated performance at the FC if suitable mitigation approaches are not employed. All these reasons could result in unreliable data at the FC from the physical sensors. In the case of humans, unreliable data could be due to the lack of skill by the human worker that could result in imperfect information from him/her. Although unintentional, the lack of knowledge has the same effect as other unreliable data and can cause a degraded performance at the FC. Sometimes, the unreliableness could also be due to the spammers in the network who provide random data as done in crowdsourcing networks where the workers are typically anonymous. Besides such scenarios, the maliciousness may also exist in some cases, where an external user gets into the task to provide intentional false data while also learning about the task. Also important in such cases is the privacy of the task.

This thesis focuses on distributed inference problems with data from potentially unreliable agents (both humans and/or sensors).

2.3 Past Work

The problem of inference from multiple agents that are potentially unreliable, is not a new one. In 1982, Lamport et al. presented the so-called *Byzantine generals problem* as follows [81]: “a group of generals of the Byzantine army camped with their troops around an enemy city. Communicating only by messenger, the generals must agree upon a common battle plan. However, one or more of them may be traitors who will try to confuse 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 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. This section reviews the Byzantine generals problem in the context of distributed inference [159], where data collected from remote locations are sent to an FC for processing and inference. The assumption is that the data are potentially tampered or falsified by some internal adversary who has the knowledge about the algorithm used at the FC. This problem has been referred to as distributed inference with Byzantine data. The Byzantine data problem in statistical inference has to take into account the inherent randomness in the data. Even without the presence of an adversary, one cannot expect perfect inference; detections can at best be correct with high probability as parameter estimates almost always are not equal to the true value. Therefore, there is a need for a probabilistic approach to the Byzantine data problem in distributed inference. Vempaty et al. provide a concise survey of results in this area [159]. 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 [65,67,90,98,117,158]. 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 [37,51,58,117,156,158].

Most of the above works deal with the case of detection. This thesis provides the first result for the case of distributed estimation with Byzantine data by considering location estimation as the estimation task. Also, the set of results presented focus on the case when the agents are sensors although the original motivation of the *Byzantine generals problem* is that of human decision makers. This thesis makes progress in that direction also by providing results for the case of distributed inference with unreliable humans using crowdsourcing as an application. The results are both information theoretic that characterize the effect of unreliable humans taking part in an inference task and also signal processing based where we propose coding theory based algorithms to ensure reliable inference from such unreliable humans. This thesis also takes steps in the direction of inference using both humans and machines simultaneously to perform complex tasks accurately and quickly.

2.4 Background Material

This section provides some mathematical background needed to understand some of the results in this thesis. Besides basic knowledge of detection and estimation theory and information and coding theory, the following algorithms are helpful.

2.4.1 Distributed Classification Fusion using Error Correcting Codes

In this subsection, we give a brief overview of Distributed Classification Fusion using Error Correcting Codes (DCFECC) approach proposed in [166]. In [166], the authors propose an M-ary distributed classification task using binary quantized data to reduce communication between the local sensors and the FC. After processing the observations locally, possibly in the presence of sensor faults, the N local sensors transmit their local decisions to the FC. In the DCFECC approach, a code matrix C is selected to perform both local decision and fault-tolerant fusion at the FC. The code matrix is an $M \times N$ matrix with elements $c_{(j+1)i} \in \{0, 1\}$, $j = 0, 1, \ldots, M-1$ and $i = 1, \ldots, N$. Each hypothesis H_j is associated with a row in the code matrix C and each column represents a binary decision rule at the local sensor. The optimal code matrix is designed off-line using techniques such as simulated annealing or cyclic column replacement [166]. After receiving the binary decisions $\boldsymbol{u} = [u_1, \ldots, u_N]$ from local sensors, the FC performs minimum Hamming distance based fusion and decides on the hypothesis H_j for which the Hamming distance between the row of C corresponding to H_j for $j = 0, \ldots, M-1$ and the received vector $\boldsymbol{u}$ is minimum. In other words, the fusion rule used by the fusion center is to decide H_j, where

$$j = \underset{0 \leq j \leq M-1}{\arg\min}\, d_H(\boldsymbol{u}, \mathbf{r}_j),$$

$d_H(\mathbf{x}, \mathbf{y})$ is the Hamming distance between vectors $\mathbf{x}$ and $\mathbf{y}$, and $\mathbf{r}_j$ is the row of C corresponding to hypothesis H_j. The tie-break rule is to randomly pick a codeword from those with the same smallest Hamming distance to the received vector. Due to the minimum Hamming-distance based fusion scheme, the DCFECC approach can also handle missing data. When an agent does not return any decision, its contribution to the Hamming distance between the received vector and every row of the code matrix is the same and therefore, the row corresponding to the minimum Hamming distance remains unchanged. The error-correction property of the code matrix provides fault-tolerance capability [166]. It is important to note that the above scheme is under the assumption that $N > M$ and the performance of the scheme depends on the minimum Hamming distance d_{min} of the code matrix C.

2.4.2 Distributed Classification using Soft-decision Decoding

In this subsection, we present a brief overview of Distributed Classification using Soft-decision Decoding (DCSD) approach proposed in [165]. This approach uses a soft-decision decoding rule as opposed to the hard-decision decoding rule used in the DCFECC approach. The use of soft-decision decoding makes the system robust to fading channels between the sensors and the FC. The basic difference between the two approaches (DCFECC and DCSD) is the decoding rule. In DCFECC, the minimum Hamming distance rule is used. In the presence of fading channels, the received data at the FC is analog although the local sensors transmit quantized data based on the code matrix C as described before. Then, the FC can use hard-decision decoding to determine the quantized data sent by the local sensors and use minimum Hamming distance rule to determine the true class. However, in [165], the authors show that the performance can deteriorate when hard-decision decoding is used. Instead, they propose a soft-decision decoding rule based on the channel statistics to determine the true class. We skip the derivation of the soft-decision decoding rule but present the decoding rule here for the case when binary quantizers are used at the local sensors, i.e., the elements of the code matrix are '0' or '1'.

Let the analog data received at the FC from the local sensors be $\boldsymbol{v} = [v_1, \cdots, v_N]$ when the local sensors transmit $\boldsymbol{u} = [u_1, \cdots, u_N]$, where $u_i = 0/1$ is decided by the code matrix C. For fading channels between the local sensors and the FC, v_i and u_i are related as follows

$$v_i = h_i(-1)^{u_i}\sqrt{E_b} + n_i, \tag{2.1}$$

where h_i is the channel gain that models the fading channel, E_b is the energy per bit and n_i is the zero mean additive white Gaussian noise. Define the reliability of the received data v_i as

$$\psi_i = \ln \frac{P(v_i|u_i = 0)P(u_i = 0|0) + P(v_i|u_i = 1)P(u_i = 1|0)}{P(v_i|u_i = 0)P(u_i = 0|1) + P(v_i|u_i = 1)P(u_i = 1|1)} \tag{2.2}$$

for $i = \{1, \cdots, N\}$. Here $P(v_i|u_i)$ can be obtained from the statistical model of the fading channel considered and $P(u_i = d|s)$ for $s, d = \{0, 1\}$ is given as follows

$$P(u_i = d|s) = \sum_{j=0}^{M-1} P(u_i = d|H_j)P_i(H_j|s). \tag{2.3}$$

$P(u_i = d|H_j)$ depends on the code matrix while $P_i(H_j|s)$ is the probability that the hypothesis H_j is true given s is present at the bit i (column i of the code matrix) before local decision making, and can be expressed as

$$P_i(H_j|s) = \frac{P_i(s|H_j)}{\sum_{l=0}^{M-1} P_i(s|H_l)} \tag{2.4}$$

where

$$P_i(s|H_l) = \begin{cases} 1, & \text{if } c_{(l+1)i} = s \\ 0, & \text{if } c_{(l+1)i} \neq s \end{cases}. \tag{2.5}$$

Then the decoding rule is to decide the hypothesis H_j where $j = \underset{0 \le j \le M-1}{\arg\min}\, d_F(\boldsymbol{\psi}, \boldsymbol{c}_{j+1})$. Here

$d_F(\boldsymbol{\psi}, \boldsymbol{c}_{j+1}) = \sum_{i=1}^{N}(\psi_j - (-1)^{c_{(j+1)i}})^2$ is the distance between $\boldsymbol{\psi} = [\psi_1, \cdots, \psi_N]$ and $(j+1)^{th}$ row of C and is referred to as F-distance.

2.5 Summary of Contributions

Table 2.1 summarizes the contributions of the thesis. The contributions of the thesis can be split into three broad parts: inference based on sensor networks, human networks, and human-machine inference networks. The results in the first two parts can be further divided into two major classes: the first which analyzes the effect of unreliable agents in the network and the second which focuses on design of the system to ensure reliable inference from these unreliable agents. The final part focuses on the development of the human-machine collaborative architecture and develops a path for future research in this area.

Table 2.1: Summary of contributions

Inference task	**Agent type**	**Cause of unreliability**	**Results**
Estimation	Sensors	Byzantines Imperfect channels	Determined optimal Byzantine attack Byzantine mitigation schemes Coding theoretic schemes
Estimation	Humans	Cognitive limitations	Non-regular CEO problem $1/R^2$ convergence rate
Detection	Humans	Bounded rationality	Decision fusion by humans Bayesian hierarchical model Sociotechnical system design
Classification	Humans	Lack of domain knowledge	Design of easy-to-answer questions Coding theoretic classification scheme
Problem-solving	Both	Bounded rationality	Machine as *coach* to human for knowledge discovery Machine as *colleague* for solution search

CHAPTER 3

ESTIMATION IN SENSOR NETWORKS: UNRELIABLE AGENTS

3.1 Introduction

In this chapter, we address the case of unreliable sensors (referred to as Byzantines) in distributed estimation. A related work was carried out for distributed detection by Marano et al. [90] and for primary user detection for cognitive radio networks by Rawat et al. in [117]. The problem of Byzantines has also been investigated in the context of network coding and information theory in [77] and [78].

The location estimation problem in wireless sensor networks (WSNs) is considered where sensors quantize their local observations before sending it to the FC. The FC estimates the location of the target using the sensors' locations and their quantized observations. In such a setup, a target localization scheme based on Monte Carlo methods is proposed for the Bayesian setup and the effect of malicious sensors is investigated. Two attack strategies are considered: independent and collaborative, and the optimal attack is derived that minimizes the posterior Fisher information or maximizes the posterior Cramér-Rao lower bound (PCRLB). The remainder of the chapter is organized as follows: In Sec. 3.2, the system model is introduced and the assumptions made are laid out. The estimation process is also developed and the performance metrics are defined. In Sec. 3.3, the performance of the estimation process is analyzed in the presence of independent attacks. The optimal strategies for both the honest and the Byzantine sensors are determined. In Sec. 3.4, collaborative attacks of Byzantines are introduced and an analysis similar to the one with independent attacks is performed. Numerical results that support our theoretical analyses of Byzantines are presented. The problem of tracking a moving target is considered in Sec. 3.5 and similar analysis is performed. Concluding discussion is provided in Sec. 3.6.

3.2 Preliminaries

3.2.1 System Model

Consider a scenario where N sensors are deployed in a WSN to estimate the location of a target present at $\theta = [x_t, y_t]$ where x_t and y_t denote the coordinates of the target location in the 2-D Cartesian plane as shown in Fig. 3.1. Although the sensors in Fig. 3.1 are shown to be deployed on a regular grid, the schemes proposed here are capable of handling any kind of sensor deployment as long as the location information for each sensor is available at the FC. Assume that the target location has a prior distribution $p_0(\theta)$. For simplicity, we assume that $p_0(\theta)$ is a Gaussian distribution, i.e., $\theta \sim \mathcal{N}(\mu, \sigma_\theta^2 \mathbf{I})$, where the mean μ is the center of a Region of Interest (ROI) and $\sigma_\theta^2 \mathbf{I}$ is very large such that the ROI includes the target's (100-t)% confidence region (typically taken to be 99%). The signal radiated from this location is assumed to follow an isotropic power attenuation model given as

$$a_i^2 = P_0 \left(\frac{d_0}{d_i}\right)^n , \tag{3.1}$$

where a_i is the signal amplitude received at the ith sensor, P_0 is the power measured at a reference distance d_0, n is the path-loss exponent, and d_i is the distance between the target and the ith sensor. Note that $d_i \neq 0$, i.e., the target is not co-located with a sensor. This assumption is valid as the probability of a target being exactly at the same location as the sensor is zero, therefore, a_i is almost surely not unbounded. Without loss of generality, fix $d_0 = 1$ and $n = 2$. The signal amplitude is corrupted by additive white Gaussian noise (AWGN) at each sensor:

$$s_i = a_i + n_i, \tag{3.2}$$

where s_i is the corrupted signal at the ith sensor and the noise n_i follows $\mathcal{N}(0, \sigma^2)$. This noise is considered to be independent across sensors. Note that the signal model given in (3.1) and (3.2) has been verified experimentally for acoustic signals in [85] and it results from averaging time samples of the received acoustic energy. The interested reader is referred to [85, 97, 126] for details.

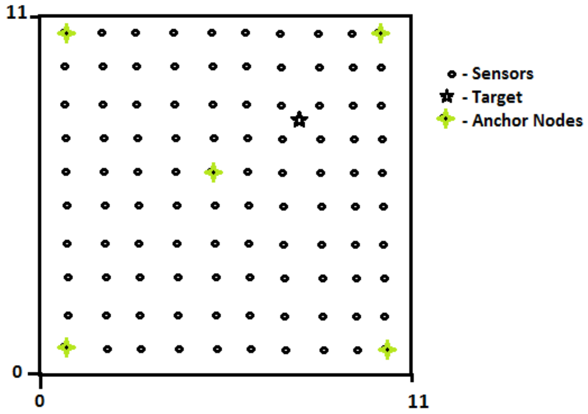


Fig. 3.1: The target in a grid deployed sensor field with anchor sensors

Due to bandwidth and energy limitations, each sensor uses a binary quantizer and sends its quantized binary measurement to the FC. The FC is assumed to know the sensor locations and the sensors use threshold quantizers because of their simplicity [119] in terms of both implementation and analysis:

$$D_i = \begin{cases} 0 & s_i < \eta_i \\ 1 & s_i > \eta_i \end{cases} \tag{3.3}$$

where D_i is the quantized binary measurement and η_i is the quantization threshold at the ith sensor. The FC receives the binary vector $\mathbf{u} = [u_1, \ldots, u_N]$ from all the sensors in the network. After collecting $\mathbf{u}$, the FC estimates the location $\theta = [x_t, y_t]$ by using the Minimum Mean Square Error

(MMSE) estimator as in [95], i.e., $\hat{\theta} = E[\theta|\mathbf{u}]$, where $E[\cdot|\mathbf{u}]$ denotes the expectation with respect to the posterior probability density function (pdf) $p(\theta|\mathbf{u})$. Since $E[\theta|\mathbf{u}]$ cannot be calculated in a closed from, we compute it using an importance sampling based Monte Carlo method described in detail below. We also assume the presence of K anchor sensors in the network similar to [95]. These anchor sensors are assumed to be secure and are used to obtain an initial estimate of the target location, θ.

3.2.2 Monte Carlo Method based Target Localization

With the recent advances in computation power, Monte Carlo based methods have become useful tools for inference problems. An importance sampling based Monte Carlo method [46] is used to approximate the posterior pdf, $p(\theta|\mathbf{u})$, as

$$p(\theta|\mathbf{u}) = \sum_{m=1}^{N_p} w_m \delta(\theta - \theta_m), \tag{3.4}$$

where the approximation is obtained as a weighted sum of N_p particles. The particles $\theta_m = [x_m, y_m]$ are drawn from the prior distribution $p_0(\theta)$. The weights are calculated using the data $\mathbf{u}$ and are proportional to the likelihood function

$$\tilde{w}_m \propto p(\mathbf{u}|\theta_m) w_m^0, \tag{3.5}$$

where the initial weights are set as identical, i.e., $w_m^0 = 1/N_p$. The updated weight of each particle is the original weight multiplied by the likelihood function of the data. Since the sensors' data are conditionally independent, we have $p(\mathbf{u}|\theta_m) = \prod_{i=1}^{N} p(u_i|\theta_m)$ for $m = 1, \ldots, N_p$. The particle weights are then normalized as

$$\begin{aligned} w_m &= \frac{\tilde{w}_m}{\sum_{m=1}^{N_p} \tilde{w}_m} && (3.6) \\ &= \frac{p(\mathbf{u}|\theta_m)}{\sum_{m=1}^{N_p} p(\mathbf{u}|\theta_m)} && (3.7) \end{aligned}$$

This results in the location estimate $\hat{\theta}$ given by

$$\hat{\theta} = \sum_{m=1}^{N_p} w_m \theta_m. \tag{3.8}$$

3.2.3 Performance Metrics

PCRLB and posterior Fisher Information Matrix (FIM) are used as the performance metrics to analyze the estimation performance [145, 146]. Let $\hat{\theta}(\mathbf{u})$ be an estimator of the target location θ. Then, the covariance matrix of the estimation error is bounded below by the PCRLB, $\mathbf{F}^{-1}$,

$$E\left\{[\hat{\theta}(\mathbf{u}) - \theta][\hat{\theta}(\mathbf{u}) - \theta]^T\right\} \geq \mathbf{F}^{-1}. \tag{3.9}$$

In (3.9), $\mathbf{F}$ is the posterior FIM [146] given as

$$\begin{aligned} \mathbf{F} &= -E_{\theta,\mathbf{u}}[\nabla_\theta \nabla_\theta^T \ln P(\mathbf{u}, \theta)] && (3.10) \\ &= -E_{\theta,\mathbf{u}}[\nabla_\theta \nabla_\theta^T \ln P(\mathbf{u}|\theta)] - E_\theta[\nabla_\theta \nabla_\theta^T \ln p_0(\theta)] && (3.11) \\ &= \mathbf{F_D} + \mathbf{F_P} && (3.12) \end{aligned}$$

where ∇_θ is the gradient operator defined as $\nabla_\theta = \left[\frac{\partial}{\partial x_t}, \frac{\partial}{\partial y_t}\right]^T$. $\mathbf{F_D}$ and $\mathbf{F_P}$ represent the contributions of data and the prior to $\mathbf{F}$ respectively. The elements of $\mathbf{F}$ are:

$$F_{11} = \int p_0(\theta) \left(\sum_{i=1}^{N} \sum_{l=0}^{1} \frac{1}{P(u_i = l|\theta)} \left[\frac{\partial P(u_i = l|\theta)}{\partial x_t} \right]^2 \right) d\theta + \frac{1}{\sigma_\theta^2}, \tag{3.13}$$

$$F_{22} = \int p_0(\theta) \left(\sum_{i=1}^{N} \sum_{l=0}^{1} \frac{1}{P(u_i = l|\theta)} \left[\frac{\partial P(u_i = l|\theta)}{\partial y_t} \right]^2 \right) d\theta + \frac{1}{\sigma_\theta^2}, \tag{3.14}$$

$$F_{21} = F_{12} = \int p_0(\theta) \sum_{i=1}^{N} \sum_{l=0}^{1} \frac{1}{P(u_i = l|\theta)} \left[\frac{\partial P(u_i = l|\theta)}{\partial x_t} \right] \left[\frac{\partial P(u_i = l|\theta)}{\partial y_t} \right] d\theta, \tag{3.15}$$

where $P(u_i = l|\theta)$ for $l = 0, 1$ is the probability that a sensor sends $u_i = l$. This value depends on Byzantines' attacking strategy.

3.3 Localization in the Presence of Byzantine Sensors

3.3.1 Independent Attack Model

For a major part of this chapter, we assume that the Byzantines attack the network independently. In an independent attack, each Byzantine sensor attacks the network by relying on its own observation without any knowledge regarding the presence of other Byzantines or their observations. Let the number of Byzantines present in the network be $M = \alpha N$. When the channels between sensors and FC are ideal, for an honest sensor, $u_i = D_i$, whereas for the Byzantines, we assume that they flip their quantized binary measurements with probability p. Therefore, under the Gaussian noise assumption, the probability that $u_i = 1$ is given by

$$P(u_i = 1|\theta, i = Honest) = Q\left(\frac{\eta_i - a_i}{\sigma}\right) \tag{3.16}$$

$$P(u_i = 1|\theta, i = Byzantine) = p\left(1 - Q\left(\frac{\eta_i - a_i}{\sigma}\right)\right) + (1-p)Q\left(\frac{\eta_i - a_i}{\sigma}\right) \tag{3.17}$$

where $Q(\cdot)$ is the complementary cumulative distribution function of a standard Gaussian distribution defined as

$$Q(x) = \frac{1}{\sqrt{2\pi}}\int_x^{\infty} e^{-\frac{t^2}{2}}dt.$$

The probability of a sensor sending a quantized value '1' is given by

$$P(u_i = 1|\theta) = (1-\alpha)Q\left(\frac{\eta_i - a_i}{\sigma}\right) + \alpha\left(p\left(1 - Q\left(\frac{\eta_i - a_i}{\sigma}\right)\right) + (1-p)Q\left(\frac{\eta_i - a_i}{\sigma}\right)\right). \tag{3.18}$$

Proposition 3.3.1. *Under independent attack, the PCRLB is given by* $\mathbf{F^{-1}}$*, where* $\mathbf{F}$ *is the posterior*

FIM whose elements are given by

$$F_{11} = \sum_{i=1}^{N} \int_{\theta} p_0(\theta) \frac{(1-2\alpha p)^2 a_i^2 e^{-\frac{(\eta_i - a_i)^2}{\sigma^2}} (x_i - x_t)^2}{2\pi\sigma^2 d_i^4 P_1(1-P_1)} + \frac{1}{\sigma_\theta^2}, \tag{3.19}$$

$$F_{12} = F_{21} = \sum_{i=1}^{N} \int_{\theta} p_0(\theta) \frac{(1-2\alpha p)^2 a_i^2 e^{-\frac{(\eta_i - a_i)^2}{\sigma^2}} (x_i - x_t)(y_i - y_t)}{2\pi\sigma^2 d_i^4 P_1(1-P_1)}, \tag{3.20}$$

$$F_{22} = \sum_{i=1}^{N} \int_{\theta} p_0(\theta) \frac{(1-2\alpha p)^2 a_i^2 e^{-\frac{(\eta_i - a_i)^2}{\sigma^2}} (y_i - y_t)^2}{2\pi\sigma^2 d_i^4 P_1(1-P_1)} + \frac{1}{\sigma_\theta^2}. \tag{3.21}$$

where $P_1 = P(u_i = 1|\theta)$ *is given in (3.18).*

Proof. The proof follows from the definition of $\mathbf{F}$ given in (3.10)-(3.15) and using the following

$$\frac{\partial P_1}{\partial x_t} = -\frac{(1-2\alpha p) a_i e^{-\frac{(\eta_i - a_i)^2}{2\sigma^2}} (x_i - x_t)}{\sigma\sqrt{2\pi} d_i^2}, \tag{3.22}$$

$$\frac{\partial P_1}{\partial y_t} = -\frac{(1-2\alpha p) a_i e^{-\frac{(\eta_i - a_i)^2}{2\sigma^2}} (y_i - y_t)}{\sigma\sqrt{2\pi} d_i^2}. \tag{3.23}$$

□

It can be observed that when $\alpha = 0$, i.e., when all the sensors are honest, the above expression simplifies to the special case of $\mathbf{F_D} = \mathbf{E}_\theta[\mathbf{J}(\theta)]$ where $\mathbf{J}(\theta)$ is the Fisher information matrix derived by Niu et al. in [101] for the case of target localization using quantized data in the absence of Byzantines.

3.3.2 Blinding the Fusion Center

The goal of Byzantines is naturally to cause as much damage to the functionality of the FC as possible. We call the event of causing the maximum possible damage as *blinding* the FC which refers to making the FC incapable of using the data from the local sensors to estimate the target location. This is clearly the case when the data's contribution to posterior Fisher Information matrix, $\mathbf{F_D}$, approaches zero. In this scenario, the best the FC can do is to use the prior to estimate the location. In other words, $\mathbf{F}$ approaches the prior's contribution to posterior Fisher Information, $\mathbf{F_P} = \frac{1}{\sigma_\theta^2}\mathbf{I}$ or the PCRLB approaches $\sigma_\theta^2\mathbf{I}$. Since PCRLB and FIM are matrix-valued and are functions of α, the blinding condition corresponds to the trace of PCRLB tending to $2\sigma_\theta^2$ or the determinant of FIM tending to $\frac{1}{\sigma_\theta^4}$. That is, α_{blind} is defined as

$$\alpha_{blind} := \min\{\alpha |\, \mathrm{tr}(\mathbf{F}(\alpha)^{-1}) = 2\sigma_\theta^2\}, \tag{3.24}$$

or

$$\alpha_{blind} := \min\left\{\alpha ||\mathbf{F}(\alpha)| = \frac{1}{\sigma_\theta^4}\right\}. \tag{3.25}$$

A closed form expression for α_{blind} can be derived and further analysis of the localization process in the presence of Byzantines can be carried out if all the honest sensors are identical and similarly all the Byzantines are identical. Therefore, in the following, we continue the analysis assuming all the honest sensors use the same local threshold η_H and all the Byzantines use η_B.

Byzantines Modeled as a Binary Symmetric Channel (BSC)

From the FC's perspective, the binary data received from the local sensor is either a false observation of the local sensor with probability $q = \alpha p$ or a true observation with probability $1 - q = (1 - \alpha) + \alpha(1 - p)$. Therefore, the effect of Byzantines, as seen by the FC, can be modeled as a Binary Symmetric Channel (BSC) with transition probability q. It is clear that this virtual 'channel' affects the PCRLB at the FC, which is a function of q. It has been shown in the literature [105] that the Cramér-Rao Lower Bound (CRLB) of the localization process approaches infinity when this transition probability approaches $\frac{1}{2}$ irrespective of the true location of the target (θ). This result means that the data's contribution to posterior Fisher Information $\mathbf{F_D}$ approaches 0 for $q = \alpha p = \frac{1}{2}$. Observe that higher the probability of flipping of the Byzantines (p), the lower the fraction of Byzantines required to blind the FC. So, the minimum fraction of Byzantines, α_{blind}, is

$\frac{1}{2}$ corresponding to $p = 1$. In order to blind the network, the Byzantines need to be at least 50% in number and should flip their quantized local observations with probability $p = 1$.

Another interpretation of this problem can be provided from the information theoretic perspective. Since the Byzantines' effect is modeled as a BSC, the capacity of this channel is $C = 1 - H_b(q)$ where $H_b(q)$ is the Binary Entropy Function given by

$$H_b(q) = -q \log_2 q - (1-q) \log_2 (1-q). \tag{3.26}$$

The FC receives non-informative data from the sensors or becomes blind, when the capacity approaches 0 which happens when $H_b(q) = 1$ or $q = \frac{1}{2}$. Following the discussion above, we have $\alpha_{blind} = \frac{1}{2}$ and $p = 1$.

It can also be observed that the data's contribution to F_{11} and F_{22} elements of $\mathbf{F}$ given by (3.19) and (3.21) become 0 when $\alpha p = \frac{1}{2}$. Once again, we get $\alpha_{blind} = \frac{1}{2}$ and $p = 1$ as the optimal attack strategy to blind the FC. Due to this observation, in the remainder of the section, we assume that the Byzantines flip their observations with probability 1, i.e., $p = 1$.

3.3.3 Best Honest and Byzantine Strategies: A Zero-Sum Game

When α, the fraction of Byzantine sensors in the network, is greater than or equal to α_{blind}, attackers will be able to blind the FC. But when α is not large enough to blind the FC, the Byzantine sensors will try to maximize the damage by making either $\mathrm{tr}(\mathbf{F}^{-1})$ as large as possible or $|\mathbf{F}|$ as small as possible. In contrast, the FC will try to minimize $\mathrm{tr}(\mathbf{F}^{-1})$ or maximize $|\mathbf{F}|$. This will result in a game between the FC and each Byzantine attacker where each player has competing goals. Each Byzantine sensor will adjust its threshold η_B to maximize $\mathrm{tr}(\mathbf{F}^{-1})$ or minimize $|\mathbf{F}|$ while the FC will adjust the honest sensor's threshold η_H to minimize $\mathrm{tr}(\mathbf{F}^{-1})$ or maximize $|\mathbf{F}|$. Thus, it is a zero-sum game where the utility of the FC is $-\mathrm{tr}(\mathbf{F}^{-1})$ (or $|\mathbf{F}|$) and the utility of the Byzantine sensor is $\mathrm{tr}(\mathbf{F}^{-1})$ (or $-|\mathbf{F}|$) [50]. More formally, let us consider $\mathrm{tr}(\mathbf{F}^{-1})$ and denote $\mathcal{C}(\eta_H, \eta_B) = \mathrm{tr}(\mathbf{F}^{-1})$ as the cost function adopted by the honest sensors. Let η_H^* and η_B^* denote the best threshold (strategy) of the honest and the Byzantine sensors, respectively. For a given η_B, η_H^* is computed as

$$\eta_H^* = \arg\min_{\eta_H} \mathcal{C}(\eta_H, \eta_B). \tag{3.27}$$

Similarly, for a given η_H, η_B^* is computed as

$$\eta_B^* = \arg\min_{\eta_B} -\mathcal{C}(\eta_H, \eta_B). \tag{3.28}$$

The solutions to (3.27) and (3.28) characterize the Nash equilibria which are defined as follows [50].

Definition 3.3.2. *A (pure) strategy η_H^* for the honest sensor is a Nash equilibrium (NE) if*

$$\mathcal{C}(\eta_H^*, \eta_B^*) \leq \mathcal{C}(\eta_H, \eta_B^*). \tag{3.29}$$

Similarly, a (pure) strategy η_B^* for the Byzantine sensor is a Nash equilibrium (NE) if

$$\mathcal{C}(\eta_H^*, \eta_B^*) \geq \mathcal{C}(\eta_H^*, \eta_B). \tag{3.30}$$

In a zero-sum game, the best strategy for both players is the saddle point, at which none of the players have the incentive to change their strategy. The saddle point for this problem given by (3.27) and (3.28) can be found using traditional methods. First, we find the set of stationary points of $\operatorname{tr}(\mathbf{F}^{-1})$ defined by $\mathcal{S}$,

$$\mathcal{S} := \left\{ (\eta_H, \eta_B) | \frac{\partial \operatorname{tr}(\mathbf{F}^{-1})}{\partial \eta_H} = \frac{\partial \operatorname{tr}(\mathbf{F}^{-1})}{\partial \eta_B} = 0 \right\}. \tag{3.31}$$

The saddle point, (η_H^*, η_B^*), is the one at which the Hessian matrix is indefinite, i.e., the determinant of the Hessian matrix is negative,

$$(\eta_H^*, \eta_B^*) = \left\{ (\eta_H, \eta_B) \in \mathcal{S} | \frac{\partial^2 \operatorname{tr}(\mathbf{F}^{-1})}{\partial^2 \eta_H} \frac{\partial^2 \operatorname{tr}(\mathbf{F}^{-1})}{\partial^2 \eta_B} - \left(\frac{\partial^2 \operatorname{tr}(\mathbf{F}^{-1})}{\partial \eta_H \partial \eta_B} \right)^2 < 0 \right\}. \tag{3.32}$$

Note that the above expressions are with respect to $\mathrm{tr}(\mathbf{F}^{-1})$. Similar analysis can be carried out when $\mathrm{tr}(\mathbf{F}^{-1})$ is replaced with $|\mathbf{F}|$ as the performance metric.

3.3.4 Numerical Results

In this subsection, we present simulation results in support of our analysis of Byzantines in a localization problem. We consider the WSN model where $N = 100$ sensors are randomly deployed in a 11×11 square region of interest where the target is located. The target's location is randomly generated from the prior $p_0(\theta)$ with $\sigma_\theta = 2.1352$ such that its 99% confidence region covers the entire ROI. There are $M = \alpha N$ Byzantine sensors present in the network who try to manipulate the data and send falsified information to the FC. We assume that the power at the reference point ($d_0 = 1$) is $P_0 = 200$. The signal amplitude at the local sensor is corrupted by AWGN with standard deviation $\sigma = 3$. In Figs. 3.2 and 3.3, we plot the values of $\mathrm{tr}(\mathbf{F}^{-1})$ and $|\mathbf{F}|$ against α in the case of an independent attack with $\eta_H = \eta_B = 8.5$. The figures show that when $\alpha = 0.5$, the $\mathrm{tr}(\mathbf{F}^{-1})$ approaches $2\sigma_\theta^2$ and $|\mathbf{F}|$ approaches $\frac{1}{\sigma_\theta^4}$. This shows that α_{blind} is equal to 1/2, i.e., unless the number of Byzantine sensors is greater or equal to 50 percent of the total number of sensors, the FC can not be made blind under independent attack. This supports our theoretical analysis regarding the PCRLB approaching $\sigma_\theta^2\mathbf{I}$ when α, which is the transition probability of the BSC model, approaches $\frac{1}{2}$. These results can be reproduced for different values of η_H and η_B. Fig. 3.4 shows the increase in Mean Square Error (MSE) of the target estimate with α. As the fraction of Byzantines increases, the MSE increases as illustrated in Fig. 3.4. Since the MSE is lower bounded by $\mathrm{tr}(\mathbf{F}^{-1})$, the plot in Fig. 3.4 is always above the plot of $\mathrm{tr}(\mathbf{F}^{-1})$ versus α in Fig. 3.2.

As discussed in Sec. 3.3.3, when $\alpha < \alpha_{blind}$, there exists a zero-sum game between the FC and Byzantine sensors, in which the optimal strategies are given by the saddle points (equilibrium points). In Figs. 3.5 and 3.6, we plot $\mathrm{tr}(\mathbf{F}^{-1})$ with varying thresholds for $\alpha = 0.4$ and we observe that there exists a saddle point (η_H^*, η_B^*) which provides optimal strategies for both types of sensors. The saddle point (η_H^*, η_B^*), in this particular example, is at $(8.5, 8.5)$. This result is intuitive as Byzantines flip their decision with probability 1. Therefore, the best strategy for the Byzantines is

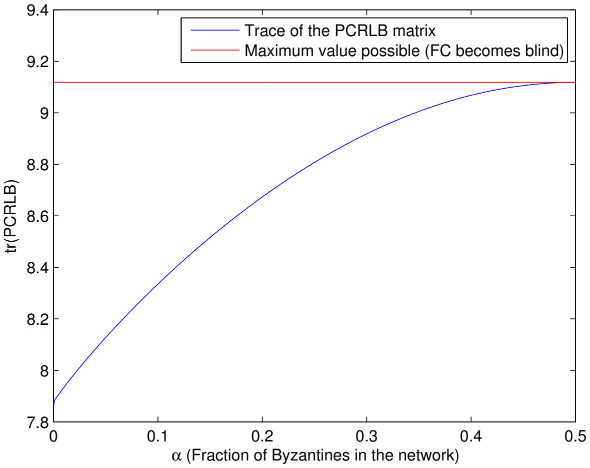


Fig. 3.2: Plot of $\mathrm{tr}(\mathbf{F}^{-1})$ versus α

to use the same best response of the Honest sensors and then flip them with probability 1. Similar results can be obtained for different values of α.

Figs. 3.7 and 3.8 show similar game theoretic analysis results for the case of $|\mathbf{F}|$ as the performance metric. The optimal values for this case (η_H^*, η_B^*) are the same $(8.5, 8.5)$. Thus, there exist saddle points (η_H^*, η_B^*) which yield the optimal strategies for both the FC and Byzantine attackers. We would like to point out that the two objective functions used in the analysis $(\mathrm{tr}(\mathbf{F}^{-1})$ and $|\mathbf{F}|)$ need not always result in the same operational point in general. However, in this particular example, this value turns out to be the same, irrespective of the performance metric.

3.4 Collaborative Attack

Next, consider the case of Byzantine attacks where all the malicious sensors communicate with each other and attack the network in a coordinated fashion. In a collaborative attack, Byzantines collaborate to deteriorate the network's estimation performance and attack the network after colluding with others. Here again, assume α to be the fraction of Byzantines present in the network. Analysis of the collaborative attack is significantly more complicated than the independent case. Here, a reasonable lower bound for α_{blind}, namely α_{blind}^L, is provided for this case.

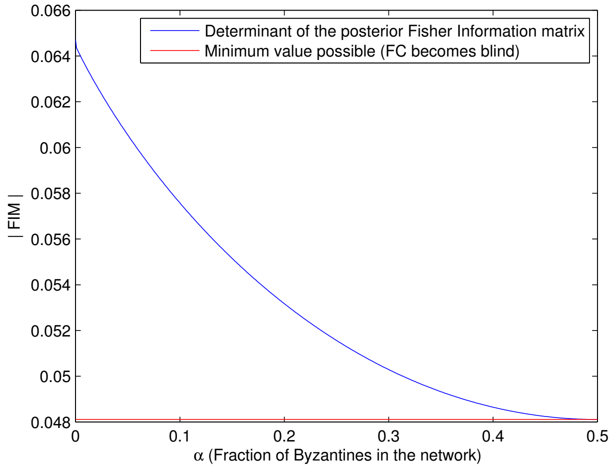


Fig. 3.3: Plot of $|\mathbf{F}|$ versus α

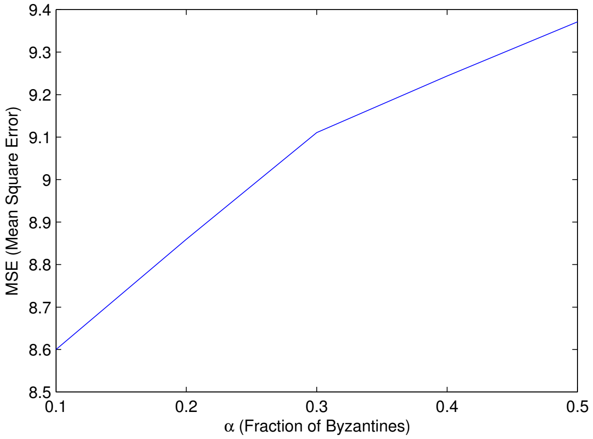


Fig. 3.4: Plot of MSE versus α

In order to find the lower bound for α_{blind}, assume that the exact location of target θ can be perfectly learned by Byzantine sensors due to collaboration. Thus, consider the case where Byzantine attackers know the location of the target and use this information collaboratively to improve their attack on the network. Let us first consider the case where each sensor uses an identical threshold. In such a case, the optimal strategy for the Byzantine sensors will be to send u_i based on the true θ value and their locations. For a given sensor i, its location $\theta_i = [x_i, y_i]$, its observation model, and its threshold $\eta_i = \eta$, the probability of the sensor sending a quantized

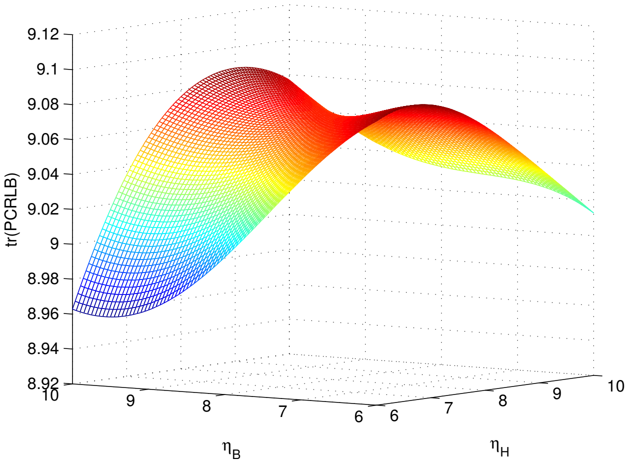


Fig. 3.5: Surface plot of $\mathrm{tr}(\mathbf{F}^{-1})$ versus honest and Byzantine sensor's threshold, η_H and η_B. The existence of a saddle point is clear.

value 1 as seen by the FC is:

$$P_i(u_i = 1|\theta) = (1-\alpha)P_i^H(u_i = 1|\theta) + \alpha P_i^B(u_i = 1|\theta) \tag{3.33}$$

The Byzantines would like to design their variables α and $P_i^B(u_i = 1|\theta)$ so that the FC becomes blind to the information received from this ith sensor, i.e., the information received from the ith sensor does not help the FC estimate the target location. This can be achieved by making the value $P_i(u_i = 1|\theta)$ a constant value (k) with respect to θ. Let $P_{i,inf}^H := \inf_\theta(P_i^H(u_i = 1|\theta))$ and the $P_{i,sup}^H := \sup_\theta(P_i^H(u_i = 1|\theta))$. Then for the ith sensor, if it was honest and $P_i^H(u_i = 1|\theta) = P_{i,inf}^H$, it would mean that the target is as far away from the ith sensor as possible. However, if the same sensor behaved as a Byzantine it is reasonable to assume that a sensible Byzantine would send a 1 to the FC with as high a probability as possible, i.e., $P_i^B(u_i = 1|\theta) = 1$. Similarly, if the ith sensor was honest and $P_i^H(u_i = 1|\theta) = P_{i,sup}^H$, then it would mean that the target is as close to the ith sensor as possible and if the same sensor behaved as a Byzantine, under a similar assumption as before, it would send a 1 to the FC with as low a probability as possible, i.e., $P_i^B(u_i = 1|\theta) = 0$.

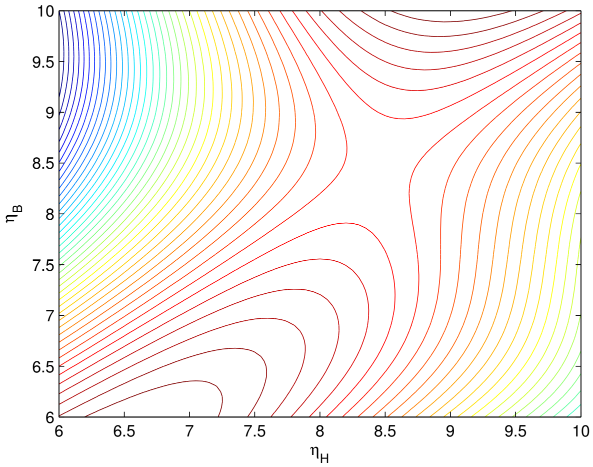


Fig. 3.6: Contour plot of the surface of $\mathrm{tr}(\mathbf{F}^{-1})$ shown in Figure 3.5.

This gives us two equations in two unknowns, $\alpha^L_{blind,i}$ and k:

$$(1-\alpha^L_{blind,i})P^H_{i,inf} + \alpha^L_{blind,i}.1 = k \tag{3.34}$$

and

$$(1-\alpha^L_{blind,i})P^H_{i,sup} + \alpha^L_{blind,i}.0 = k. \tag{3.35}$$

After solving (3.34) and (3.35), we get

$$\alpha^L_{blind,i} = \frac{P^H_{i,sup} - P^H_{i,inf}}{1 + P^H_{i,sup} - P^H_{i,inf}} \tag{3.36}$$

where $\alpha^L_{blind,i}$ is the fraction of malicious sensors required to make sensor i non-informative to the FC. In this case, we have the following at the FC,

$$P_i(u_i = 1|\theta) = (1-\alpha^L_{blind,i})P^H_i(u_i = 1|\theta) + \alpha^L_{blind,i}P^B_i(u_i = 1|\theta) = \frac{P^H_{i,sup}}{1 + P^H_{i,sup} - P^H_{i,inf}} \tag{3.37}$$

which is a constant independent of θ. Thus, when $\alpha \geq \alpha^L_{blind,i}$ for a particular honest sensor i, the attackers can have $\alpha - \alpha^L_{blind,i}$ fraction of Byzantine sensors act like honest sensors and have the

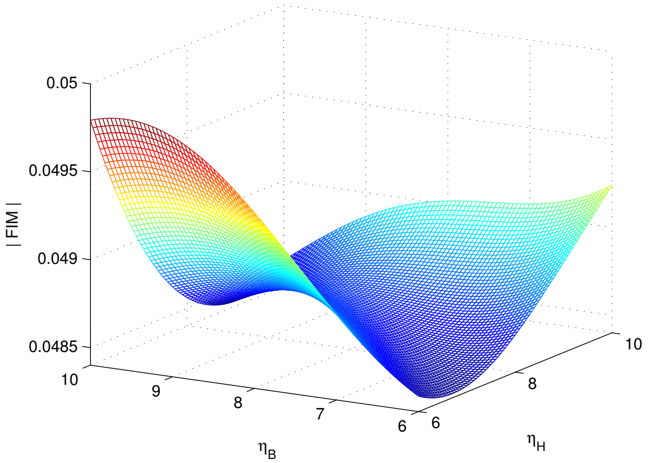


Fig. 3.7: Plot of $|\mathbf{F}|$ versus honest and Byzantine sensor's threshold, η_H and η_B. The existence of a saddle point is clear.

rest $\alpha^L_{blind,i}$ Byzantine sensors send 1 with probability as below:

$$P^B_i(u_i = 1|\theta) = \frac{P^H_{i,sup} - P^H_i(u_i = 1|\theta)}{P^H_{i,sup} - P^H_{i,inf}}$$

This causes the FC to become incapable of utilizing the received information from the ith sensor to estimate the target location. In order to guarantee that the FC cannot obtain any useful information from any of its sensors, the minimum required $\alpha^{L,ident}_{blind}$ is given as

$$\alpha^{L,ident}_{blind} = \max_i \alpha^L_{blind,i}. \tag{3.38}$$

This provides us with a lower bound, $\alpha^{L,ident}_{blind}$, for the collaborative case under the identical threshold scheme. For the collaborative attack case, it can be observed that $\alpha^L_{blind,i}$ given by (3.36) is always ≤ 0.5 which implies that $\alpha^{L,ident}_{blind} \leq 0.5$ which is the α_{blind} obtained in the independent attack case. This shows that if a strategy exists to obtain this lower bound, then the fraction of sensors required to blind the FC would decrease in the collaborative attack case as compared to the independent attack case. A similar observation was made by Rawat et al. in [117] for the primary user detection for cognitive radio networks in the presence of Byzantines.

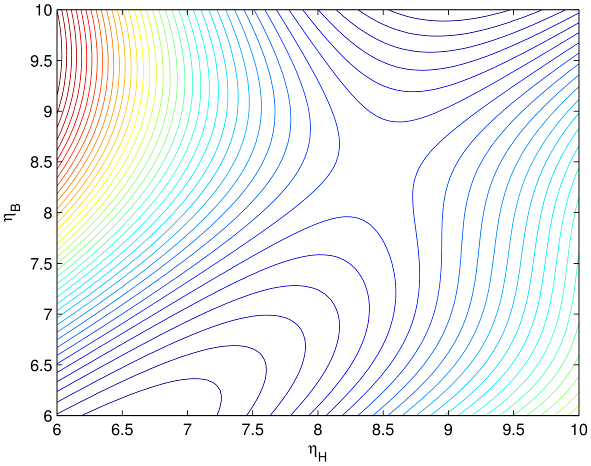


Fig. 3.8: Plot of contour of the surface of $|\mathbf{F}|$ shown in Fig. 3.7.

3.5 Target Tracking in the Presence of Byzantine Sensors

Now, we analyze the effect of Byzantines on a more general estimation problem namely a target tracking problem in WSNs. We first discuss the system model and the performance metrics relevant to the tracking framework. Using a similar Byzantine attack model as used for the location estimation task, we evaluate the effect of Byzantines on the tracking performance.

3.5.1 System Model

We consider a single target moving in a two-dimensional Cartesian coordinate plane whose dynamics is defined by the 4-dimensional state vector $\theta_{\mathbf{k}} = [x_{tk}\ y_{tk}\ \dot{x}_{tk}\ \dot{y}_{tk}]^T$ where x_{tk} and y_{tk} denote the x and y coordinates of the target respectively, at time k. $\dot{x}_{tk}$ and $\dot{y}_{tk}$ denote the first order derivatives (velocities) in x and y directions, respectively. Target motion is defined by the white noise acceleration model [13] as described below:

$$\theta_{\mathbf{k}} = \mathbf{F}\theta_{\mathbf{k-1}} + \nu_k, \tag{3.39}$$

where

$$\mathbf{F} = \begin{bmatrix} 1 & 0 & T & 0 \\ 0 & 1 & 0 & T \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{3.40}$$

and ν_k is the additive white Gaussian process noise which is assumed to be zero mean with covariance matrix $\mathbf{Q}$ given by

$$\mathbf{Q} = q \begin{bmatrix} \frac{T^3}{3} & 0 & \frac{T^2}{2} & 0 \\ 0 & \frac{T^3}{3} & 0 & \frac{T^2}{2} \\ \frac{T^2}{2} & 0 & T & 0 \\ 0 & \frac{T^2}{2} & 0 & T \end{bmatrix}, \tag{3.41}$$

where q and T denote the process noise parameter and the time interval between adjacent sensor measurements, respectively. We assume that the FC has an exact knowledge of the target state-space model (3.39) and the process noise statistics. There are N sensors deployed in the network. The dynamic target radiates a signal which is assumed to follow an isotropic power attenuation model same as the location estimation problem:

$$a_{ik}^2 = P_0 \left(\frac{d_0}{d_{ik}} \right)^n, \tag{3.42}$$

where a_{ik} is the signal amplitude received at the ith sensor at time instant k, P_0 is the power measured at a reference distance d_0, n is the path-loss exponent, and d_{ik} is the distance between the target and the ith sensor at the kth time step. Without loss of generality, we assume $d_0 = 1$ and $n = 2$. The signal amplitude is assumed to be corrupted by additive white Gaussian noise (AWGN) at each sensor:

$$s_{ik} = a_{ik} + n_i, \tag{3.43}$$

where s_{ik} is the corrupted signal at the ith sensor at time instant k and the noise n_i follows $\mathcal{N}(0, \sigma^2)$. We assume this noise to be independent across sensors. Due to energy and bandwidth constraints, each sensor quantizes its received signal s_{ik} locally using a binary quantizer and the quantized data is sent to the FC. We consider threshold quantizers for their simplicity in terms of implementation and analysis:

$$v_{ik} = \begin{cases} 0 & s_{ik} < \eta_{ik} \\ 1 & s_{ik} > \eta_{ik} \end{cases} . \tag{3.44}$$

In (3.44), v_{ik} is the locally quantized binary measurement of the ith sensor and η_{ik} is the quantization threshold used by this sensor at time instant k. The FC receives the binary vector $\mathbf{u_k} = [u_{1k}, \cdots, u_{Nk}]$ from all the sensors in the network, where u_{ik} may not be equal to v_{ik} due to the presence of Byzantine sensors in the network (c.f. 3.5.3). After collecting $\mathbf{u_k}$, the FC sequentially estimates the target state θ_k using a sequential importance resampling (SIR) particle filter [11].

3.5.2 Performance Metrics

As used before for the location estimation problem, we use PCRLB as the metric for tracking performance. Let $\hat{\theta}_\mathbf{k}(\mathbf{u_{1:k}})$ be an estimator of the state vector $\theta_\mathbf{k}$ at time k, given all the available measurements $\mathbf{u_{1:k}} = [\mathbf{u_1} \cdots \mathbf{u_k}]$ up to time k. Then, the mean square error (MSE) matrix of the estimation error at time k, is bounded below by the PCRLB $\mathbf{J_k}^{-1}$ [45],

$$\mathbf{B_k} = E\left\{\left[\hat{\theta}_\mathbf{k}\left(\mathbf{u}_{1:k}\right) - \theta_\mathbf{k}\right]\left[\hat{\theta}_\mathbf{k}\left(\mathbf{u}_{1:k}\right) - \theta_\mathbf{k}\right]^T\right\} \geq \mathbf{J_k}^{-1}, \tag{3.45}$$

where $\mathbf{J_k}$ is the Fisher information matrix (FIM). Tichavský *et al.* in [143] provide a recursive approach to calculate this sequential FIM $\mathbf{J_k}$:

$$\mathbf{J_{k+1}} = \mathbf{D_k^{22}} - \mathbf{D_k^{21}}\left(\mathbf{J_k} + \mathbf{D_k^{11}}\right)^{-1}\mathbf{D_k^{12}}, \tag{3.46}$$

where

$$\begin{aligned}
\mathbf{D_k^{11}} =& E\left\{-\nabla_{\theta_\mathbf{k}}\nabla^T_{\theta_\mathbf{k}} \log\, p\left(\theta_\mathbf{k+1}|\theta_\mathbf{k}\right)\right\}, && (3.47)\\
\mathbf{D_k^{12}} =& E\left\{-\nabla_{\theta_\mathbf{k}}\nabla^T_{\theta_\mathbf{k+1}} \log\, p\left(\theta_\mathbf{k+1}|\theta_\mathbf{k}\right)\right\}, && (3.48)\\
\mathbf{D_k^{21}} =& E\left\{-\nabla_{\theta_\mathbf{k+1}}\nabla^T_{\theta_\mathbf{k}} \log\, p\left(\theta_\mathbf{k+1}|\theta_\mathbf{k}\right)\right\} = \left(\mathbf{D_k^{12}}\right)^T, && (3.49)\\
\mathbf{D_k^{22}} =& E\left\{-\nabla_{\theta_\mathbf{k+1}}\nabla^T_{\theta_\mathbf{k+1}} \log\, p\left(\theta_\mathbf{k+1}|\theta_\mathbf{k}\right)\right\}\\
&+ E\left\{-\nabla_{\theta_\mathbf{k+1}}\nabla^T_{\theta_\mathbf{k+1}} \log\, p\left(\mathbf{u}_{k+1}|\theta_\mathbf{k+1}\right)\right\}\\
=& \mathbf{D_k^{22,a}} + \mathbf{D_k^{22,b}}. && (3.50)
\end{aligned}$$

The derivative operator $\nabla_{\theta_\mathbf{k}}$ in (3.47)-(3.50) is defined as

$$\nabla_{\theta_k} = \left[\frac{\partial}{\partial x_{tk}}, \frac{\partial}{\partial y_{tk}}, \frac{\partial}{\partial \dot{x}_{tk}}, \frac{\partial}{\partial \dot{y}_{tk}}\right]^T \tag{3.51}$$

and the expectations in (3.47)-(3.50) are taken with respect to the joint probability distribution $p\left(\theta_\mathbf{0:k+1}, \mathbf{u}_{1:k+1}\right)$. The *a priori* probability density function (pdf) of the target state $p_0(\theta_\mathbf{0})$ can be used to calculate the initial FIM as $\mathbf{J_0} = E\left\{-\nabla_{\theta_\mathbf{0}}\nabla^T_{\theta_\mathbf{0}} \log\, p_0\left(\theta_\mathbf{0}\right)\right\}.$

For the target dynamic model and the measurement model used in this paper, the expressions in (3.47)-(3.50) simplify to

$$\begin{aligned}
\mathbf{D_k^{11}} &= \mathbf{F}^T\mathbf{Q}^{-1}\mathbf{F}, && (3.52)\\
\mathbf{D_k^{12}} &= \left(\mathbf{D_k^{21}}\right)^T = -\mathbf{F}^T\mathbf{Q}^{-1,} && (3.53)\\
\mathbf{D_k^{22}} &= \mathbf{Q}^{-1} + \mathbf{D_k^{22,b}}. && (3.54)
\end{aligned}$$

Note that $\mathbf{D_k^{22,b}}$ is the only term that depends on the observations $\mathbf{u}_{1:k}$ of the local sensors.

3.5.3 Attack Model

In this section, we analyze the system in the presence of Byzantines. For this target tracking problem, we focus on independent attacks only. Since the quantized observations transmitted by the local sensors are binary in nature, the Byzantines can attack the network by changing this binary quantized data. Let $M = \alpha N$ be the number of Byzantines in the network. It is assumed that the FC knows the fraction of Byzantines (α) but does not the know the identity of the Byzantines. An honest sensor sends its true observation to the FC and therefore, $u_{ik} = v_{ik}$, whereas for the Byzantines, we assume that they flip their quantized binary measurements with probability p. Therefore, under the Gaussian noise assumption, the probability that u_{ik} = 1 is given by

$$P(u_{ik} = 1|\theta_{\mathbf{k}}, i = Honest) = Q\left(\frac{\eta_{ik} - a_{ik}}{\sigma}\right), \tag{3.55}$$

$$P(u_{ik} = 1|\theta_{\mathbf{k}}, i = Byzantine) = p\left(1 - Q\left(\frac{\eta_{ik} - a_{ik}}{\sigma}\right)\right) + (1-p)Q\left(\frac{\eta_{ik} - a_{ik}}{\sigma}\right), \tag{3.56}$$

where $Q(\cdot)$ is the complementary cumulative distribution function of a standard Gaussian distribution. From the FC's perspective, the probability of a sensor sending '1' is, therefore, given by

$$P_{ik} = P(u_{ik} = 1|\theta_{\mathbf{k}}) = (1-\alpha)Q\left(\frac{\eta_{ik} - a_{ik}}{\sigma}\right) + \alpha\left(p\left(1 - Q\left(\frac{\eta_{ik} - a_{ik}}{\sigma}\right)\right) + (1-p)Q\left(\frac{\eta_{ik} - a_{ik}}{\sigma}\right)\right). \tag{3.57}$$

Proposition 3.5.1. *Under independent Byzantine attacks, data's contribution* $\mathbf{D}_{\mathbf{k}}^{\mathbf{22,b}}$ *to FIM is given by:*

$$\mathbf{D}_{\mathbf{k}}^{\mathbf{22,b}} = \begin{bmatrix} {\mathbf{D}_{\mathbf{k}}^{\mathbf{22,b}}}_{11} & {\mathbf{D}_{\mathbf{k}}^{\mathbf{22,b}}}_{12} & 0 & 0 \\ {\mathbf{D}_{\mathbf{k}}^{\mathbf{22,b}}}_{21} & {\mathbf{D}_{\mathbf{k}}^{\mathbf{22,b}}}_{22} & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}, \tag{3.58}$$

where the non-zero elements are given by

$$\mathbf{D_k^{22,b}}_{11} = \sum_{i=1}^{N} E\left[\frac{(1-2\alpha p)^2 a_{il}^2 e^{-\frac{(\eta_{il}-a_{il})^2}{\sigma^2}}(x_{il}-x_{tl})^2}{2\pi\sigma^2 d_{il}^4 P_{il}(1-P_{il})}\right], \tag{3.59}$$

$$\mathbf{D_k^{22,b}}_{12} = \mathbf{D_k^{22,b}}_{21} = \sum_{i=1}^{N} E\left[\frac{(1-2\alpha p)^2 a_{il}^2 e^{-\frac{(\eta_{il}-a_{il})^2}{\sigma^2}}(x_{il}-x_{tl})(y_{il}-y_{tl})}{2\pi\sigma^2 d_{il}^4 P_{il}(1-P_{il})}\right], \tag{3.60}$$

$$\mathbf{D_k^{22,b}}_{22} = \sum_{i=1}^{N} E\left[\frac{(1-2\alpha p)^2 a_{il}^2 e^{-\frac{(\eta_{il}-a_{il})^2}{\sigma^2}}(y_{il}-y_{tl})^2}{2\pi\sigma^2 d_{il}^4 P_{il}(1-P_{il})}\right]. \tag{3.61}$$

where for $l = k+1$*,* P_{il} *is given in* (3.57) *and the expectation is taken with respect to* $p\left(\theta_{\mathbf{0:k}}, \mathbf{u}_{\mathbf{1:k}}\right) p\left(\theta_{\mathbf{k+1}}|\theta_{\mathbf{k}}\right)$.

Proof. Note that the only non-zero terms of $\mathbf{D_k^{22,b}}$ are in the 2×2 sub-matrix $\mathbf{D_k^{22,b}}(1:2,1:2)$. This is due to the fact that the distribution of data does not depend on the first order derivatives ($\dot{x}_{tk}$ and $\dot{y}_{tk}$) of the state $\theta_{\mathbf{k}}$. Based on the fact that $\theta_{\mathbf{k}}$, $\theta_{\mathbf{k+1}}$ and $\mathbf{u}_{k+1}$ form a Markov chain, the joint PDF for the expectation can be rewritten as follows

$$p\left(\theta_{\mathbf{0:k+1}}, \mathbf{u}_{1:k+1}\right) = p\left(\theta_{\mathbf{0:k}}, \mathbf{u}_{1:k}\right) p\left(\theta_{\mathbf{k+1}}|\theta_{\mathbf{k}}\right) p\left(\mathbf{u}_{k+1}|\theta_{\mathbf{k+1}}\right). \tag{3.62}$$

The remainder of the proof follows from the definition of $\mathbf{D_k^{22,b}}$ given in (3.50) and using the following

$$\frac{\partial P_{i(k+1)}}{\partial x_{t(k+1)}} = -\frac{(1-2\alpha p) a_{i(k+1)} e^{-\frac{\left(\eta_{i(k+1)}-a_{i(k+1)}\right)^2}{2\sigma^2}}\left(x_{i(k+1)}-x_{t(k+1)}\right)}{\sigma\sqrt{2\pi} d_{i(k+1)}^2}, \tag{3.63}$$

$$\frac{\partial P_{i(k+1)}}{\partial y_{t(k+1)}} = -\frac{(1-2\alpha p) a_{i(k+1)} e^{-\frac{\left(\eta_{i(k+1)}-a_{i(k+1)}\right)^2}{2\sigma^2}}\left(y_{i(k+1)}-y_{t(k+1)}\right)}{\sigma\sqrt{2\pi} d_{i(k+1)}^2}. \tag{3.64}$$

□

3.5.4 Blinding the Fusion Center

The goal of Byzantines is naturally to cause as much damage to the functionality of the FC as possible. Following the same terminology as used for the location estimation problem, we call the event of causing maximum damage as *blinding* the FC which refers to making the data from the local sensors non-informative for the FC. In our case, this happens when the data's contribution $\mathbf{D_k^{22,b}}$ to FIM, approaches zero. In this scenario, the best the fusion center can do is to use the information from the prior knowledge of state transition model to estimate the state. Since PCRLB/FIM is matrix-valued and it is a function of α, we define the blinding condition to mean that the trace of $D_k^{22,b}$ is zero. That is, α_{blind} may be defined as

$$\alpha_{blind} \triangleq \min\{\alpha | \operatorname{tr}(\mathbf{D_k^{22,b}}(\alpha)) = 0\}. \tag{3.65}$$

For our framework, a closed form expression for α_{blind} can be derived and further analysis of target tracking in the presence of Byzantines can be carried out by carefully observing the expressions derived in Prop. 3.5.1. Observe that the non-zero elements of $\mathbf{D_k^{22,b}}$, data's contribution to FIM, given by (3.59)-(3.61) all become zero when $\alpha p = \frac{1}{2}$, i.e., $\mathbf{D_k^{22,b}} = \mathbf{0}$ when $\alpha p = \frac{1}{2}$. This implies that the higher the probability of flipping of Byzantines (p), the lower the fraction of Byzantines required to blind the FC from local sensors' data. Therefore, the minimum fraction of Byzantines, α_{blind}, is $\frac{1}{2}$ corresponding to $p = 1$. Byzantines need to be at least 50% in number and should flip their quantized local observations with probability '1' to blind the network.

3.6 Discussion

In this chapter, the problems of target localization and target tracking with Byzantine sensors have been considered and the effect of unreliable agents (sensors) has been explored. Optimal attacking strategies have been theoretically found along with the fundamental limits of attack that corresponds to no information at the FC. In the following chapter, the problems are analyzed from the network's perspective to determine the approaches that the network can adopt to ensure reliable inference using unreliable sensors when the fraction of sensors is below the *blinding* fraction.

CHAPTER 4

ESTIMATION IN SENSOR NETWORKS: RELIABLE INFERENCE

4.1 Introduction

In the previous chapter, the problem of localization 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 estimation task at the FC. It was shown that when the fraction of Byzantines in the network are greater than 0.5 and are attacking independently, the FC becomes blind to the data from the sensors and can only estimate the location of the target using the prior information. However, addressing the problem from the network's perspective, one can develop techniques to counter the effect of Byzantines and ensure reliable estimation performance. In this chapter, such schemes are explored. Note that the focus is only on the independent attack case. As discussed briefly at the end of the previous chapter, the analysis for the collaborative attack case is more complex and is not considered here. Three schemes are proposed in this chapter: Byzantine identification scheme, design of dynamic non-identical threshold scheme, and coding-theoretic target localization. The Byzantine identification scheme proposed herein is similar in principle to the one proposed in [156] for a distributed detection problem where the Byzantines are identified in an adaptive fashion and their information is used adaptively to improve the system's detection performance. The dynamic non-identical threshold scheme explores the design of local quantizers by moving away from the traditional static identical thresholds to dynamic non-identical thresholds that make the system robust to Byzantines. The effectiveness of the proposed dynamic non-identical threshold scheme against Byzantines is also shown for the target tracking problem discussed in Sec.3.5. Lastly, the coding-theoretic scheme builds on the DCFECC and DCSD approaches discussed in Sec. 2.4 to develop a computationally more efficient scheme than the maximum likelihood scheme of the previous chapter. This coding-theoretic scheme is also more robust to Byzantine attacks.

The remainder of the chapter is organized as follows: In Sec. 4.2, the Byzantine identification scheme is described and validated with some numerical results. Dynamic non-identical quantizers are developed in Sec. 4.3 using variational calculus to improve the performance of the system in the presence of Byzantines. Following these Byzantine mitigation schemes, in Sec. 4.4, a new localization scheme is developed using coding theory that is computationally more efficient than the traditional maximum likelihood approach. Its performance analysis is first characterized in the absence of Byzantines and is shown to be asymptotically optimal when the number of sensors approach infinity. The performance in the presence of Byzantines is then analyzed in Sec. 4.5 and it is shown to be robust to the Byzantine attacks, both analytically and via simulations. Further analysis is provided in Sec. 4.6 where the scheme is extended to the case of non-ideal channels between the sensors and the FC. The chapter is summarized with some discussion in Sec. 4.7.

4.2 Byzantine Identification

The first scheme proposed for the mitigation of independent Byzantine attacks is to identify the Byzantines by observing their behavior over time. In the preceding chapter, we have shown how the optimal identical thresholds are designed in the presence of Byzantines based on the PCRLB for the target location estimation error. It was assumed that all the Byzantines use an identical threshold η_B and all the honest sensors use an identical threshold η_H. It can be seen in the numerical results presented in Sec. 3.3.4 that the optimal strategy for the Byzantines and the honest sensors is to use $\eta_H = \eta_B = \eta$. Here, we propose a scheme to identify the Byzantines present in the network. This scheme is similar to previous work in [156] for the cooperative spectrum sensing problem.

The basic idea of this identification scheme is to observe a sensor's behavior over time and decide on whether it behaves closer to an honest or a Byzantine sensor [156]. This is done by comparing the observed values of $\hat{\gamma}_i = P(u_i = 1|\theta)$ to the expected values of $\hat{\gamma}_i^H = P(u_i = 1|\hat{\theta}, i = Honest)$ or $\hat{\gamma}_i^B = P(u_i = 1|\hat{\theta}, i = Byzantine)$. $\hat{\gamma}_i = P(u_i = 1|\theta)$ is estimated in an iterative manner where $\hat{\gamma}_i$ at the $(T+1)$th iteration is calculated as

$$\hat{\gamma}_i(T+1) = \frac{T\hat{\gamma}_i(T) + u_i(T+1)}{T+1}. \tag{4.1}$$

The values of $\hat{\gamma}_i^H$ and $\hat{\gamma}_i^B$ can be calculated using (3.16) and (3.17). It is important to observe here that these values require the location θ which is unknown. These values in our scheme are initialized by using a coarse estimate of the location, $\hat{\theta}$. In order to obtain an initial coarse estimate, $\hat{\theta}$, a procedure similar to the one proposed by Masazade et al. in [95] is adopted. In this procedure, it is assumed that there are K anchor sensors in the network that have a higher level of security and thereby treated as honest sensors. The initial data is collected at the FC (at time $T = 0$) from these K anchor sensors and the MMSE estimate is obtained using the procedure described in Section 3.2.2. For the remainder of this section, the following model is assumed. At every iteration of the algorithm, the sensors send their 1-bit data using the pre-designed identical threshold value. Using these N sensors' data of previous T time instants, the FC iteratively updates $\hat{\gamma}_i(T+1)$.

The estimate $\hat{\gamma}_i(T)$ is computed at every iteration T and a sensor is declared honest or Byzantine based on the test statistic

$$\Lambda_i(T) = \left| \frac{\hat{\gamma}_i(T) - \hat{\gamma}_i{}^H}{\hat{\gamma}_i(T) - \hat{\gamma}_i{}^B} \right| \tag{4.2}$$

which is the ratio of the deviations between the estimated behavior of the ith sensor and the expected behavior of an honest sensor, to the estimated behavior of the ith sensor and the expected behavior of a Byzantine sensor. The FC declares a sensor as a Byzantine at time instant T if $\Lambda_i(T)$

is greater than 1. This would mean that the sensor behaves closer to a Byzantine than an honest sensor. The advantage of this scheme is that it is adaptive such that the sensor's declaration regarding the sensor being honest or Byzantine is based on data from previous time instants. It is important to note that the above formulation (4.2) is purely heuristic and no optimality is claimed. It is a sub-optimal and an easy to implement formulation. The rationale behind such a formulation is that in traditional classification/pattern recognition problems, the decision regarding the type (of a sensor) is made by observing the behavior (of the sensor). A sensor is declared as Type A, if it behaves closer to the expected behavior of Type A. In our problem, the behavior is characterized by $\hat{\gamma}_i(T)$ and a decision is made by comparing the closeness of this behavior to the expected behavior (${\hat{\gamma}_i}^H$ or ${\hat{\gamma}_i}^B$). It is important to note that, in this section, we propose a Byzantine identification scheme but do not discuss the estimation procedure. The estimation is done after a final decision is made regarding the identity of the Byzantines. The time instant when a final decision is made depends on the particular scenario and is a design criterion. Once, the final decision is made, the data from the sensors identified as Byzantines can be re-flipped and used in the estimation process similar to the adaptive fusion rule designed in [156] for the problem of distributed spectrum sensing in the presence of Byzantines. In the following sub-section, we show with numerical simulations that, for our particular example, most of the Byzantines can be identified in around 100 iterations using the proposed Byzantine identification scheme.

4.2.1 Numerical Results

The effectiveness of our identification scheme is presented through numerical results. For this scenario, the same network with $N = 100$ sensors uniformly deployed in a 11×11 area as shown in Fig. 3.1 is considered. The fraction of the Byzantines is $\alpha = 0.4$, i.e., 40 out of 100 sensors are malicious. It is assumed that there are K anchor sensors as shown in Fig. 3.1. Each sensor measures s_i, the signal amplitude a_i corrupted by AWGN with $\sigma = 3$. The power at the reference distance ($d_0 = 1$) is $P_0 = 200$. The Byzantines and honest sensors use thresholds $\eta_H = \eta_B = 8.5$, which are the optimum thresholds found in Sec. 3.3.4 assuming that the prior distribution of the target location is a normal distribution such that the region of interest (ROI) includes 99% confidence region. Each Byzantine flips its binary observation with probability 1, before sending it to the FC. The results of the proposed Byzantine identification scheme for a specific target location realization can be seen in Figs. 4.1 and 4.2. In Fig. 4.1, the number of wrongly identified sensors (an honest sensor wrongly identified as a Byzantine and vice-versa) is plotted as a function of time for different values of K, the number of anchor sensors. The value of K has been varied to observe the effect of the *coarseness* of the estimate on the Byzantine identification scheme. The graph shows that most of the sensors are correctly identified with time. The number of wrongly identified sensors is maximum for $K = 3$ and minimum for $K = 9$ as expected since the coarse estimate is more accurate when the number of anchor sensors is larger. For $K = 5$, the number of wrongly identified sensors converges to the value of 14 (out of a total of 100 sensors). From this figure, it can be inferred that $K = 5$ is a reasonable number of anchor sensors to be used. In Fig. 4.2, the estimated α given by $\alpha_{est} = \frac{\hat{M}}{N}$, where $\hat{M}$ is the number of sensors identified as Byzantines, is plotted as a function of time for a network with $K = 5$ anchor sensors placed in star formation as shown in Fig. 3.1. As can be seen from Fig. 4.2, α_{est} converges to the value of 0.42. From these figures, it can be inferred that 8 honest sensors (out of 60 honest sensors in the network) have been falsely identified as Byzantines and 6 Byzantines (out of 40 Byzantines present in the network) have been mis-identified as honest sensors.

It is important to note that the performance of the proposed scheme depends on the MSE of the initial coarse estimate. In all the simulations performed, the MSE of the initial coarse estimate is < 3 square units and in Table 4.1, we show the effect of MSE of the estimate on the performance of the identification scheme when $K = 5$ anchor sensors are used.

4.2.2 Discussion

The proposed scheme does not depend on the fraction of Byzantines (α) present in the network and, therefore, the scheme performs well for all possible values of α. It detects most of the Byzantines in the network. A major limitation of this scheme is that some sensors can not be identified reliably

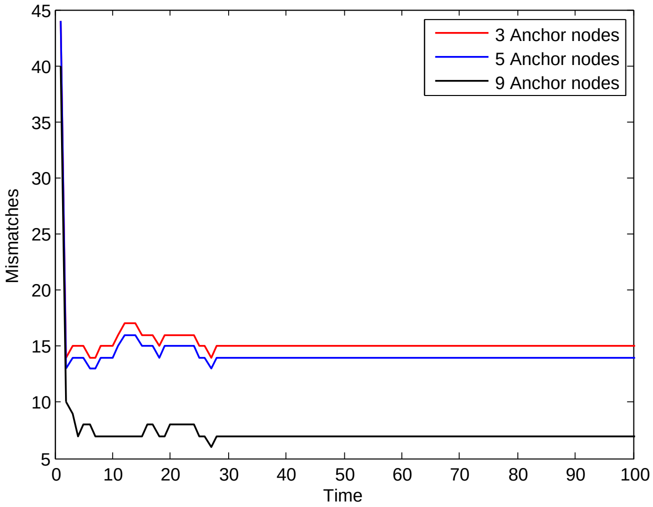


Fig. 4.1: Number of wrongly identified sensors with time

Table 4.1: Mismatches versus MSE of the initial coarse estimate found using $K = 5$ anchor sensors

MSE of initial coarse estimate	Number of Mismatches
0.4	1
0.6	2
1.4	7
1.6	8
1.9	10
2.28	12
2.44	13
2.6	14

as Byzantine or honest. It can be observed that the sensors for which this scheme fails are the sensors for which the test statistic Λ_i^T is close to 1. This happens when γ_i^H or γ_i^B is close to 0.5. These are the sensors for which $a_i = \eta$ corresponding to some constant $d_i = d$. These sensors lie on the boundary region as shown in Fig. 4.3. These $ambiguous$ sensors evade the identification process due to the following reason. An $ambiguous$ sensor is defined as the sensor i, for which the quantization threshold is approximately equal to its received amplitude a_i. Therefore, for the Gaussian noise model assumed here, $P(u_i = 1|\hat{\theta}) = \hat{\gamma}_i \approx 0.5$ for an $ambiguous$ sensor irrespective of whether it is an honest sensor or a Byzantine. This implies that these sensors send

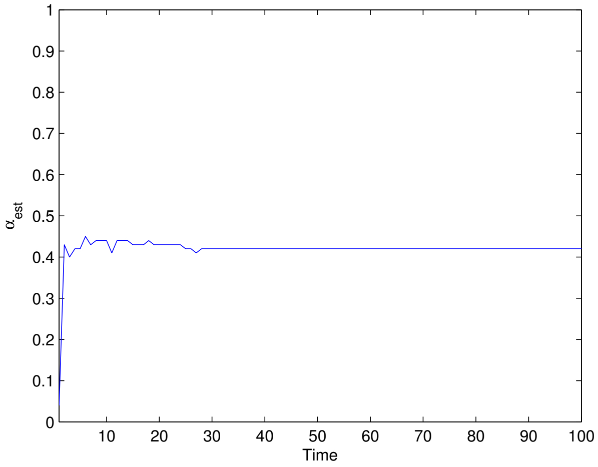


Fig. 4.2: Estimate of α with time

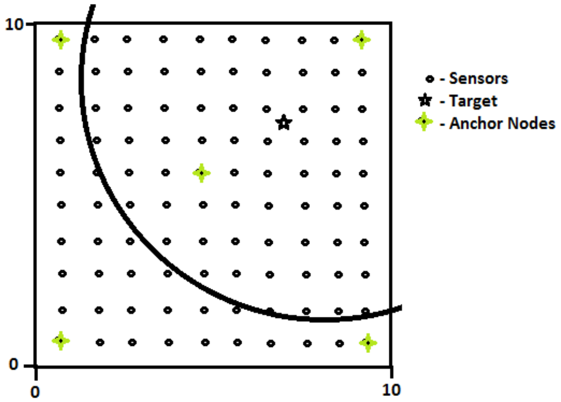


Fig. 4.3: Boundary around which the sensors are 'ambiguous'

1 with approximate probability 0.5. Since, the Byzantines flip their decision with probability 1, it cannot be inferred if it was an honest sensor genuinely sending 1 with probability 0.5 or a Byzantine sensor sending 1 with probability 0.5 after flipping its decision. Furthermore, since the received amplitude a_i is constant for $ambiguous$ sensors, they happen to form a circle around the true target location. This 'hard' decision regarding the type of sensor results in a high probability of misidentification of the sensors in the region shown in Fig. 4.3. These sensors can, therefore, be categorized as $ambiguous$. The ratio of the number of these *ambiguous* sensors on the boundary region to the total number of sensors in a network may be small in practice. In this case, they have negligible impact on estimation performance. However, this number depends on the relative positioning of the target with respect to the local sensors and on the threshold used for quantization at the local sensors. The width of the uncertain zone also depends on sensor noise variance. If this ratio is high, then the estimation performance might be severely degraded. Therefore, we need to design a scheme where the information from these sensors can be utilized for localization. This problem of boundary sensors can be alleviated if we use non-identical quantizers discussed in the following section, in conjunction with the identification scheme.

4.3 Design of Dynamic Non-Identical Quantizers

In this section, we introduce our non-identical quantizer design scheme to tackle the ambiguity caused by boundary sensors when identical quantizers are used. The estimation model in this section is sequential and described as follows. At every iteration T, all the N sensors send their one-bit data regarding the location of the target using their local thresholds. At time $T = 0$, the local sensors use the optimal identical thresholds designed in Sec. 3.3.3 for quantization. The FC estimates the target location at every time instant T using Monte-Carlo based MMSE estimation described in Sec. 3.2.2 and broadcasts this estimate information to the local sensors. The essential difference between this new scheme and the previous one is the feedback between the FC and the local sensors. In other words, according to this new scheme, local sensors update their quantizers based on the feedback information (location estimate) they receive from the FC. In order to understand the design, we first investigate the case where there are no Byzantines and all the sensors are honest.

4.3.1 Honest Sensors Only

Niu et al. in [101] analyzed the location estimation problem and proposed a threshold design method by minimizing the CRLB on the location estimation error, where the optimal thresholds are found by

$$\min_{\bar{\eta}} V(\bar{\eta}|\theta), \tag{4.3}$$

where $V(\bar{\eta})$ is the trace of the CRLB matrix. In this work, since PCRLB is the performance metric, the optimization problem can be written as

$$\min_{\bar{\eta}} \int V(\bar{\eta}|\theta) p_0(\theta) d\theta. \tag{4.4}$$

This minimization problem is a non-convex vector minimization problem over N variables. Here, this problem can be simplified by making every threshold a function of signal amplitude at the sensor and assuming that all the sensors follow the same functional dependence, i.e,

$$\eta_i = \eta(a_i), \tag{4.5}$$

where $\eta(\cdot)$ is some function and a_i is the amplitude of the observation at the ith sensor. Since the value of a_i is not known, each threshold is updated iteratively as

$$\eta_i^{T+1} = \eta(\hat{a}_i^T), \tag{4.6}$$

where $\hat{a}_i^T$ is the expected amplitude at the previous time instant which is estimated by using the location estimate of the Tth iteration, $\hat{\theta}^T$. The minimization problem now becomes a variational minimization problem

$$\min_{\eta(\cdot)} \int V(\eta(\cdot)|\theta) p_0(\theta) d\theta. \tag{4.7}$$

This is still a difficult problem as V is a function of the target's location θ which is unknown. Therefore, a heuristic approach is proposed that is similar to the one used in [101] which is explained next.

Heuristic Approach for Non-Identical Quantizer Design

It is important to observe that all the required information about the target location $\theta = [x_t, y_t]$ is completely available in the signal amplitudes a_i's. Therefore, intuitively, if one can accurately estimate a_i from u_i for $i = 1, 2, \ldots, N$, then the target location can be accurately estimated. At any given sensor, this estimation problem is to estimate a using the log-likelihood function given by

$$\ln P(u|a) = (1-u)\ln P(u=0|\eta(\hat{a}), a) + u \ln P(u=1|\eta(\hat{a}), a) \tag{4.8}$$

where a is the signal amplitude at that sensor, $\hat{a}$ is the estimate of the amplitude and $u \in \{0,1\}$ is the corresponding quantized bit-value.

Proposition 4.3.1. *The posterior Fisher Information about the signal amplitude, a, at a given sensor is given by*

$$G[\eta(\cdot)] = \int_A F[\eta(\hat{a}), a] p_A(a) da + \Gamma, \tag{4.9}$$

where $p_A(a)$ is the pdf of the signal amplitude a at the sensor and

$$F[\eta(\hat{a}), a] = \frac{e^{-\frac{(\eta(\hat{a})-a)^2}{\sigma^2}}}{2\pi\sigma^2[Q(\frac{\eta(\hat{a})-a}{\sigma})][1-Q(\frac{\eta(\hat{a})-a}{\sigma})]} \tag{4.10}$$

is the data's contribution to posterior FI and Γ is a constant representing prior's contribution to the posterior FI which is given by

$$\Gamma = -E\left[\frac{d^2 \ln p_A(a)}{da^2}\right]. \tag{4.11}$$

Proof. The proof is given in Appendix A.1. □

Now the threshold function $\eta(\cdot)$ can be designed such that it maximizes the posterior Fisher information.

Proposition 4.3.2. *The posterior Fisher Information $G[\eta(\cdot)]$ is maximized when the threshold function is $\eta(a) = a$.*

Proof. Since the second term on the right hand side of (4.9) is not a function of η, we can only consider the first term. The maximization of the first term with respect to $\eta(\cdot)$ is a functional maximization problem and it can be solved using the Euler-Lagrange equation from variational calculus [147] given by

$$\frac{\partial F}{\partial \eta} = \frac{d}{da}\frac{\partial F}{\partial \eta^{(1)}}, \tag{4.12}$$

where $\eta^{(1)}$ is the first derivative of η with respect to a. Since $F[\eta(\hat{a}), a]$ is independent of $\eta^{(1)}$, the Euler-Lagrange equation reduces to $\frac{\partial F}{\partial \eta} = 0$ or $F =$ constant with respect to η. From (4.10), this gives the result that $\eta(a) = a$ for F to be a constant with respect to η.[1] □

Such an analysis was also carried out numerically by Ribeiro et al. in [119], where the authors plotted the CRLB against $(\eta - a)$ to show that $\eta(a) = a$ minimizes the CRLB or in other words maximizes the Fisher information.

Therefore, the thresholds are designed such that, $\eta_i^{T+1} = \eta(\hat{a}_i^T) = \hat{a}_i^T$ which means that the threshold of the ith sensor at time $(T+1)$ is the estimated amplitude at this sensor at the previous time instant T. This amplitude is estimated by using the previous time instant's location estimate, $\hat{\theta}^T$, which is broadcast by the FC to the local sensors. It is important to note that this result is expected as such a threshold design will ideally yield the maximum entropy as it results in $P(u_i = 1|\theta) = P(u_i = 0|\theta) = \frac{1}{2}$.

4.3.2 Game Between Honest and Byzantine Sensors

The situation changes when there are honest sensors as well as Byzantine sensors present in the system. Since the Byzantines' aim is to deteriorate the system performance, they do not necessarily use the threshold design specified by the FC. Instead, Byzantines use their own threshold function $\eta^B(\cdot)$ and they flip their decisions with probability p. Let the threshold function of the honest sensors be $\eta^H(\cdot)$.

Proposition 4.3.3. *The posterior Fisher Information about the signal amplitude, a, at a given sensor in the presence of Byzantines is given by*

$$G[\eta^H(\cdot), \eta^B(\cdot)] = \int_A F[\eta^H(.), \eta^B(.), a] p_A(a) da + \Gamma, \tag{4.13}$$

---

[1] Interested reader is referred to Sec. 2.2 of [147] for further information on Euler-Lagrange Equation and Variational Calculus.

where the constant Γ *is given in* (4.11)*,* $p_A(a)$ *is the pdf of the signal amplitude* a *at the sensor, and*

$$F[\eta^H(\cdot),\eta^B(\cdot),a] = \frac{\left(-\alpha(2p-1)e^{-\frac{(\eta^B(\hat{a})-a)^2}{2\sigma^2}} + (1-\alpha)e^{-\frac{(\eta^H(\hat{a})-a)^2}{2\sigma^2}}\right)^2}{2\pi\sigma^2[P_1][1-P_1]}, \tag{4.14}$$

with P_1 *is defined as the probability of the sensor sending* 1 *as seen by the FC*

$$P_1 = \alpha\left(p\left(1-Q\left(\frac{\eta^B(\hat{a})-a}{\sigma}\right)\right) + (1-p)Q\left(\frac{\eta^B(\hat{a})-a}{\sigma}\right)\right) + (1-\alpha)Q\left(\frac{\eta^H(\hat{a})-a}{\sigma}\right). \tag{4.15}$$

Proof. The proof is given in Appendix A.2. ☐

In this case, the problem can again be modeled as a zero-sum game where the objective of the FC is to maximize the posterior Fisher Information $G[\eta^H(.),\eta^B(.)]$ whereas the objective of the Byzantine sensor is to minimize $G[\eta^H(.),\eta^B(.)]$. This problem can be solved by examining the expression of G in (4.13). Under the scenario that each sensor behaves independently, it can be shown that the Fisher Information (FI) given by (4.14) is maximized when honest sensors set their thresholds as $\eta^H(\hat{a}) = \hat{a}$ regardless of the value of η^B. For Byzantines, there are two cases to be considered: $\alpha p < \frac{1}{2}$ and $\alpha p \geq \frac{1}{2}$. For $\alpha p < \frac{1}{2}$, the Byzantines, who try to minimize this posterior FI, achieve minimization similarly by setting $\eta^B(\hat{a}) = \hat{a}$ regardless of the value of η^H. This result is expected as the Byzantines flip their observations with a probability p. It is important to observe that when $\eta^B(\hat{a}) = \eta^H(\hat{a}) = \hat{a}$, it implies that the honest sensors send $0/1$ with probability approximately equal to $\frac{1}{2}$. Also, observe that if the Byzantines also use this thresholding scheme, the probability of a Byzantine sending a 1 is

$$\begin{aligned} P(u=1|a, Byzantine) = p\left(1-Q\left(\frac{\eta^B(\hat{a})-a}{\sigma}\right)\right) \\ + (1-p)Q\left(\frac{\eta^B(\hat{a})-a}{\sigma}\right), \end{aligned} \tag{4.16}$$

which becomes $\frac{1}{2}$, when $\eta^B(\hat{a}) = \hat{a}$, irrespective of the value of p. Similar to the honest sensors, the Byzantines maximize the entropy as well and eventually benefit the network in the localization task as discussed earlier. This result can also be interpreted by using the BSC modeling of Byzantines. In this model, the honest sensors try to transmit the data (0/1) such that the capacity is achieved. The capacity is achieved when the input follows uniform distribution. Our threshold scheme makes $P(u_i = 0/1|\theta) = \frac{1}{2}$, thereby achieving capacity of this 'Byzantine' BSC. For $\alpha p \geq \frac{1}{2}$, the Byzantines need to use $\eta^B(\hat{a}) = \pm\infty$, that is, always send a 0 or 1. However, in this case, the network would again eventually overcome the actions of Byzantines as the FC can easily identify the Byzantines using the Byzantine identification scheme proposed in Sec. 4.2.

It is worthwhile to point out here that no final strategies are proposed for the Byzantines here. Instead, it is shown that any non-honest strategy used by the Byzantines can be identified by the FC and therefore, the effect of Byzantines can be mitigated. Therefore, utilizing dynamic non-identical quantizers ensures that the Byzantines become 'ineffective' in their attack strategy and the network eventually mitigates the actions of Byzantines. It is also important to observe that this particular framework ensures that the network is robust for any fraction (α) of the Byzantines in the system. The trade-off is that the FC needs to broadcast the target location estimate at each iteration which increases the system complexity and consume more system resources compared to using static identical quantizers. Static thresholds are set only once in the beginning and they stay constant throughout the estimation process. In the identification procedure, the FC tries to identify sensors based on their observed behavior over time, which requires each sensor to send their decisions to the FC in a continuous manner. In contrast, dynamic thresholds are adjusted dynamically at each sensor using the feedback from the FC. As one would expect, feedback not only improves estimation performance but it also makes the network more robust to Byzantine attacks.

4.3.3 Numerical Results

The superiority of the proposed dynamic non-identical quantizer design over the static identical quantizer design is now shown via simulations. For this scenario, consider a network with $N = 100$

sensors uniformly deployed in a 11×11 area same as before (refer to Fig. 3.1). The power at the reference distance ($d_0 = 1$) is again set as $P_0 = 200$. This set-up yielded an optimal identical threshold as $\eta = 8.5$. During the first iteration, the location $\theta^{(1)}$ is estimated using identical thresholds. After obtaining $\hat{\theta}^{(1)}$, the estimates in the next iterations are calculated by using the proposed dynamic non-identical threshold design scheme as $\eta_i^{T+1} = \hat{a}_i^T$. MMSE estimators are implemented using the importance sampling method as described in Sec. 3.2.2 with $N_p = 10000$ particles. Fig. 4.4 shows the mean squared error (MSE) values of the estimators for 1000 Monte-Carlo realizations of θ compared to the MSE of those using identical thresholds. As can be seen from Fig. 4.4, the estimation error reduces significantly (by around 70% in 3 iterations) when non-identical dynamic threshold quantizers are used as compared to the identical threshold quantizers. This motivates the honest sensors to use the designed non-identical thresholds. As discussed above, the Byzantines are ineffective in this proposed scheme.

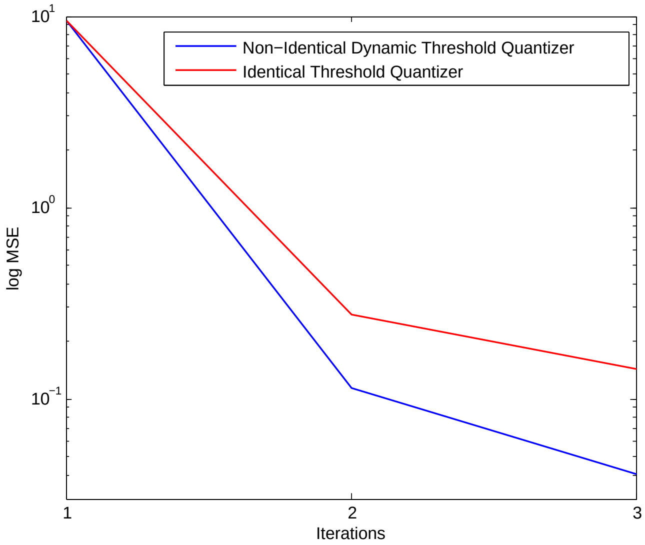


Fig. 4.4: MSE comparison of the two schemes: identical threshold scheme and non-identical dynamic threshold scheme

When each sensor uses this dynamic non-identical quantizer design scheme for the case of collaborative attack discussed in Sec. 3.4, the analysis is extremely difficult, however, the following can be conjectured. The largest deviation in the current estimate caused by the Byzantines is limited to the confidence interval of the previous estimate since, otherwise, they would be easily identified as outliers at the FC. This limits the attacking power of each Byzantine and hence, more number of Byzantines are required to cause the same blinding effect at the FC. Therefore, we conjecture that $\alpha_{blind}^{L,non-ident} \geq \alpha_{blind}^{L,ident}$, where $\alpha_{blind}^{L,non-ident}$ denotes the fraction of Byzantines required to blind the FC under dynamic non-identical quantizer scheme.

4.3.4 Target Tracking

Next, we consider the problem of target tracking and show how the non-identical threshold design scheme proposed above for the location estimation problem can also be used for the target tracking problem. The estimation model in this section is sequential and described as follows. At time k = 0, the local sensors send quantized data using the optimal identical quantizer thresholds proposed in [101]. The FC estimates the target state at every time instant using particle filtering and broadcasts this estimate information to the local sensors. At every subsequent iteration k, all the N sensors send their one-bit quantized observations using their updated local thresholds. In this new scheme, local sensors update their quantizers based on the feedback information (state estimate) they receive from the FC. In order to develop insights, we follow a similar methodology as done for location estimation problem, and investigate the case where there are no Byzantines and all the sensors are honest.

Honest Sensors Only

Note that the recursive Fisher information is a function of sensor thresholds from time 1 to time k. Using the target dynamic model in (3.39) and the sensor measurement model in (3.43), one can design optimal static threshold values offline by minimizing a cost function, which could be the trace or the determinant of the recursive PCRLB in (3.46), over the sensor thresholds. However, as was previously shown in [104], the performance can be improved using a dynamic optimal quantizer design. For the dynamic quantizers, quantizer design must not only be based on the system model but must also exploit the feedback mechanism from the FC to the sensors. In order to dynamically design thresholds in real time, one needs to take into account the information contained in the measurements up to time k, i.e., $\mathbf{u_{1:k}}$. It was also shown in [104] that this optimization is complicated and researchers have found ways to simplify the optimization problem. Here, we simplify this problem by making every local sensor threshold a function of signal amplitude at the sensor and assuming that all the sensors follow the same functional dependence, i.e,

$$\eta_{ik} = \eta(a_{ik}), \tag{4.17}$$

where $\eta(\cdot)$ is some function and a_{ik} is the amplitude of the observation at the ith sensor at time k. Since the value of a_{ik} is not known, we update each threshold iteratively as

$$\eta_{i(k+1)} = \eta(\hat{a}_{ik}), \tag{4.18}$$

where $\hat{a}_{ik}$ is the estimated amplitude at the previous time instant which is estimated by using the state estimate of the kth iteration, $\hat{\theta}_{\mathbf{k}}$. The minimization problem now becomes a variational minimization problem

$$\min_{\eta(\cdot)} \operatorname{tr}\left(\mathbf{D}_{\mathbf{k}}^{\mathbf{22,b}}(\eta(\cdot))\right). \tag{4.19}$$

This is still a difficult problem as the objective function depends on the target's true state $\theta_{\mathbf{k+1}}$ which is an unknown. Therefore, we use the heuristic approach used before for the location estimation problem in Sec. 4.3.1 to design the quantizers. This gives the threshold value as $\eta_{i(k+1)} = \eta(\hat{a}_{ik}) = \hat{a}_{ik}$ which means that the threshold of the ith sensor at time $(k+1)$ is the estimated amplitude at this sensor at the previous time instant k. This previous time instant's state estimate $\hat{\theta}_k$, is needed to estimate the amplitude, which is broadcast by the FC to the local sensors.

The analysis when there are Byzantines in the network is similar to the game-theoretic problem discussed in Sec. 4.3.2. This problem consists of two players and can be modeled as a zero-sum game. The two players, the FC and the Byzantines, have opposing objectives where the objective of the FC is to maximize the posterior Fisher Information $G[\eta^H(\cdot), \eta^B(\cdot)]$ whereas the objective of the Byzantine sensors is to minimize $G[\eta^H(\cdot), \eta^B(\cdot)]$. This problem has been solved in Sec. 4.3.2 by examining the expression of G. When sensors behave independently, the Fisher information (FI) given by (4.14) is maximized when honest sensors set their thresholds as $\eta^H(\hat{a}) = \hat{a}$ regardless of the value of η^B. For $\alpha p \leq \frac{1}{2}$, the Byzantines, who try to minimize this posterior FI, achieve the minimization similarly by setting $\eta^B(\hat{a}) = \hat{a}$ regardless of the value of η^H. This result is expected as the Byzantines flip their observations with a probability p. Observe that if the Byzantines use this thresholding scheme, the probability of a Byzantine sending a '1' is

$$P(u = 1|a, Byzantine) = p\left(1 - Q\left(\frac{\eta^B(\hat{a}) - a}{\sigma}\right)\right) + (1-p)Q\left(\frac{\eta^B(\hat{a}) - a}{\sigma}\right) \tag{4.20}$$

which becomes $\frac{1}{2}$, when $\eta^B(\hat{a}) \approx a$, irrespective of the value of p. Also, since $\eta^H(\hat{a}) \approx a$, it implies that the honest sensors also send $0/1$ with probability approximately equal to $\frac{1}{2}$.

Simulation Results

We present simulation results to show the effectiveness of our proposed dynamic non-identical threshold scheme in the presence of Byzantines for the target tracking problem. Consider a network of N sensors deployed in a grid over a 200 m $\times$ 200 m area. The sensor density, defined as the number of sensors per unit area, is denoted by ρ. The target is assumed to emit power $P_0 = 25000$ and the local observations at the sensors are assumed to be corrupted by AWGN with zero mean and variance $\sigma^2 = 1$. The target state dynamics is modeled as follows: the initial state distribution is assumed to be Gaussian with mean $\mu_\mathbf{0} = [-80\ -80\ 2\ 2]^T$ and covariance $\mathbf{\Sigma_0} = \text{diag}[10^2\ 10^2\ .5^2\ .5^2]$, the target motion model is assumed to be a near constant velocity model and the process noise parameter is $q = 0.16$. The total observation time duration is 60s and it is assumed that the observations are made every $T = 1$s. For particle filtering, we use $N_p = 1000$ particles. As we have shown that the optimal strategy for the Byzantines is to flip their local observations with probability '1' for which the $\alpha_{blind} = 0.5$, we consider the case when the fraction of Byzantines $\alpha \leq 0.5$.

The identical threshold scheme uses constant thresholds $\eta_{ik} = 1.7$ for $i = 1, \cdots, N$ and

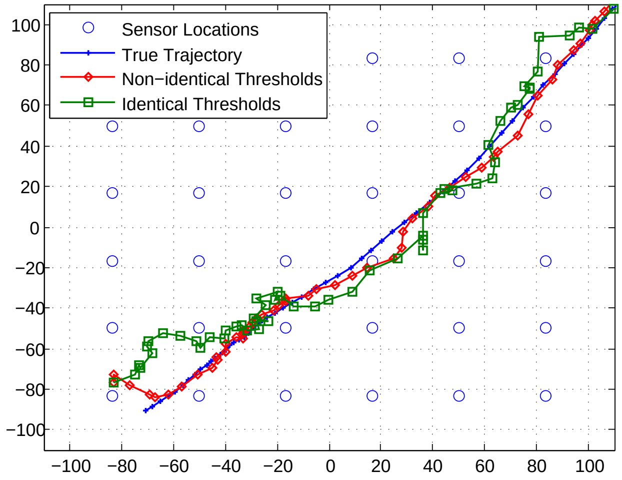


Fig. 4.5: Example of tracking using the two schemes when $\alpha = 0.1$

$k = 1, \cdots, 60$ which is the optimal static threshold designed in [101]. The dynamic non-identical threshold scheme follows the update mechanism proposed in this section. Figs. 4.5 and 4.6 show the improvement in tracking performance when the non-identical threshold scheme is used instead of the identical threshold scheme. For $\alpha = 0.1$, Fig. 4.5 shows the estimated tracks and the true track for a particular realization. It can be observed that the estimation error for both the identical and non-identical threshold schemes is not very different. However, when $\alpha = 0.3$, as seen from Fig. 4.6, the error is significantly reduced when the non-identical threshold scheme is used over the identical threshold scheme.

We now compare the average performance of both the schemes characterized by the average root mean square error (RMSE) over 100 Monte-Carlo runs. Fig. 4.7 shows this comparison of the two schemes for two different values of N: $N = 36$ ($\rho = 9 \times 10^{-4}$) and $N = 100$ ($\rho = 25 \times 10^{-4}$). As Fig. 4.7 shows, the proposed scheme performs better and it is more robust than the identical threshold scheme in the presence of Byzantines. However, note that both the schemes have the same performance when $\alpha = \alpha_{blind} = 0.5$, since $\alpha = 0.5$ means that the FC is blind to sensor data.

So far, two schemes have been presented to mitigate the effect of Byzantines in the network.

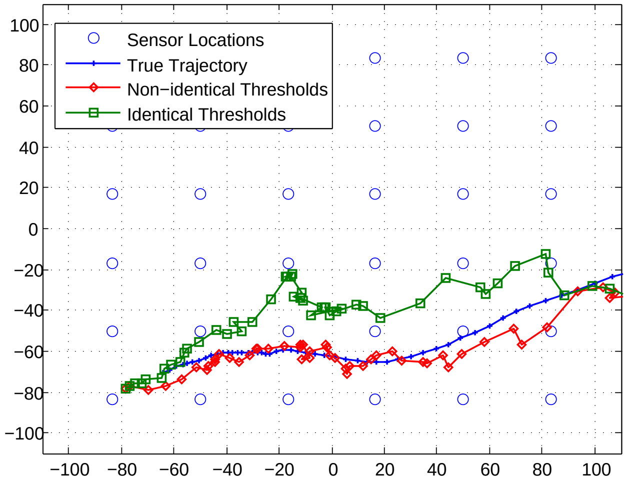


Fig. 4.6: Example of tracking using the two schemes when $\alpha = 0.3$

While the first scheme is passive and only learns the Byzantines' identity over time, the second scheme changes the system and uses a dynamic non-identical threshold scheme. However, both the schemes still use maximum likelihood or MMSE based optimal estimators. The third scheme proposed next, changes this approach and proposes a sub-optimal but easy to implement localization scheme which could be extended to include the tracking problem but is not included in this thesis. This scheme works on the idea that classification is easier than estimation and therefore, formulates the localization problem as hierarchical classification using error-correcting codes. First, the scheme is designed in the absence of any Byzantines and then the scheme is analyzed in the presence of such malicious sensors and/or imperfect channels. Note that a major change is the transition from a Bayesian framework using MMSE based estimator to the framework that uses a sub-optimal estimator.

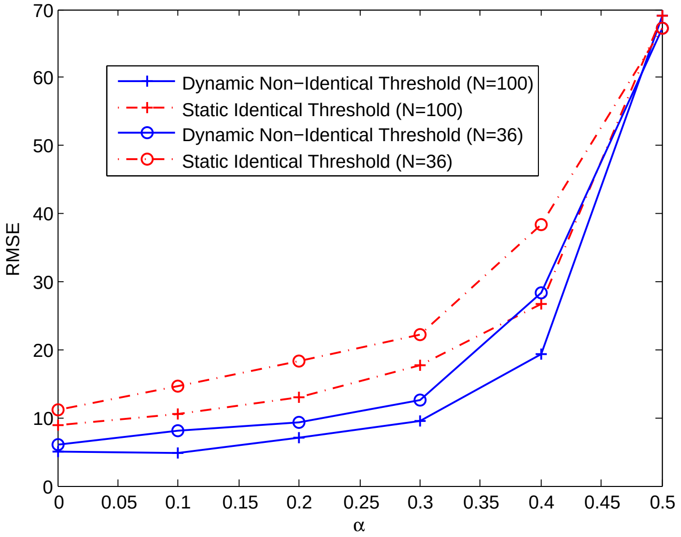


Fig. 4.7: Tracking comparison of the two schemes in the presence of Byzantines in terms of RMSE for different values of N

4.4 Localization as Hierarchical Classification

In this section, a localization scheme is proposed which is based on hierarchical classification. The algorithm is iterative in which at every iteration, an M-ary hypothesis test is performed at the FC and accordingly the ROI is split into M regions. The FC, through feedback, declares this region as the ROI for the next iteration. The M-ary hypothesis test solves a classification problem where each sensor sends binary quantized data based on a code matrix C. The code matrix is of size $M \times N$ with elements $c_{(j+1)i} \in \{0, 1\}$, $j = 0, 1, \cdots, M-1$ and $i = 1, \cdots, N$, where each row represents a possible region and each column i represents ith sensor's binary decision rule. After receiving the binary decisions $\boldsymbol{u} = [u_1, u_2, \cdots, u_N]$ from local sensors, the FC performs minimum Hamming distance based fusion. In this way, the search space for target location is reduced at every iteration and the search is stopped based on a pre-determined stopping criterion. The optimal splitting of the ROI at every iteration depends on the topology of the network and the distribution of sensors in the network. For a given network topology, the optimal region split can be determined offline using k-means clustering [18] which yields Voronoi regions [12] containing equal number of sensors in every region. For instance, when the sensors are deployed in a uniform grid, the optimal splitting is uniform as shown in Fig. 4.8. In the remainder of the chapter, we consider a symmetric sensor deployment such as a grid. Such a deployment results in a one-to-one correspondence between sensors across regions which is required in our derivations. Further discussion is provided in the later part of this section. In this section, the sensors are assumed to be benign and the channels between the local sensors and the FC are assumed to be ideal. Therefore, in this section, the binary decisions received at the FC are the same as the binary decisions made by the local sensors, i.e., $u_i = D_i$, for $i = 1, \cdots, N$ where D_i is defined in (3.3). These assumptions are relaxed in the later sections. The FC estimates the target location using the received data $\boldsymbol{u}$.

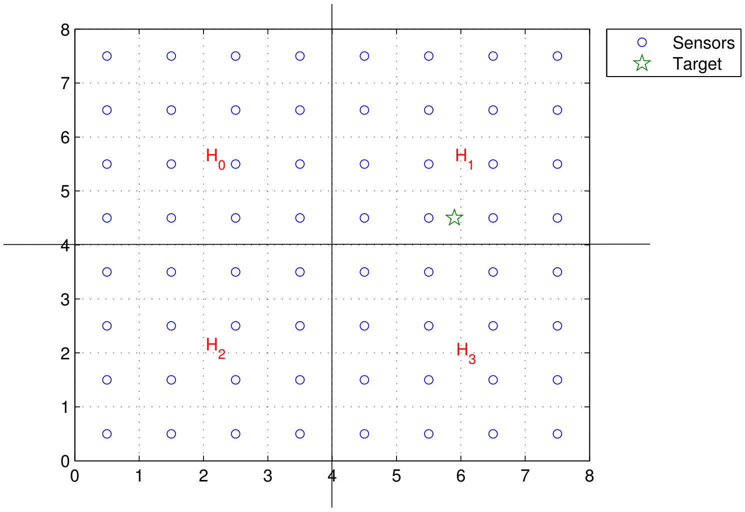


Fig. 4.8: Equal region splitting of the ROI for localization as a 4-hypothesis test

4.4.1 Basic Coding Based Scheme

In this subsection, the basic coding based scheme is presented for target localization. Since there are N sensors which are split into M regions, the number of sensors in the new ROI after every iteration is reduced by a factor of M. After k iterations, the number of sensors in the ROI are $\frac{N}{M^k}$ and therefore, the code matrix at the $(k+1)$th iteration would be of size $M \times \frac{N}{M^k}$.[1] Since the code matrix should always have more columns than rows, $k^{stop} < \log_M N$, where k^{stop} is the number of

---

[1] It is assumed that N is divisible by M^k for $k = 0, 1, \ldots, \log_M N - 1$.

iterations after which the scheme terminates. After k^{stop} iterations, there are only $\frac{N}{M^{k^{stop}}}$ sensors present in the ROI and a coarse estimate $\hat{\theta} = [\hat{\theta}_x, \hat{\theta}_y]$ of the target's location can be obtained by taking an average of locations of the $\frac{N}{M^{k^{stop}}}$ sensors present in the ROI:

$$\hat{\theta}_x = \frac{M^{k^{stop}}}{N} \sum_{i \in ROI_{k^{stop}}} x_i \tag{4.21}$$

$$\text{and} \quad \hat{\theta}_y = \frac{M^{k^{stop}}}{N} \sum_{i \in ROI_{k^{stop}}} y_i, \tag{4.22}$$

where $ROI_{k^{stop}}$ is the ROI at the last step.

Since the scheme is iterative, the code matrix needs to be designed at every iteration. Observing the structure of the problem, the code matrix can be designed in a simple and efficient way as described below. As pointed out before, the size of the code matrix C^k at the $(k+1)$th iteration is $M \times \frac{N}{M^k}$, where $0 \leq k \leq k^{stop}$. Each row of this code matrix C^k represents a possible hypothesis described by a region in the ROI. Let R_j^k denote the region represented by the hypothesis H_j for $j = 0, 1, \cdots, M-1$ and let S_j^k represent the set of sensors that lie in the region R_j^k. Also, for every sensor i, there is a unique corresponding region in which the sensor lies and the hypothesis of the region is represented as $r^k(i)$. It is easy to see that $S_j^k = \{i \in ROI_k | r^k(i) = j\}$. The code matrix is designed in such a way that for the jth row, only those sensors that are in R_j^k have a '1' as their elements in the code matrix. In other words, the elements of the code matrix are given by

$$c_{(j+1)i}^k = \begin{cases} 1 & \text{if } i \in \mathcal{S}_j^k \\ 0 & \text{otherwise} \end{cases}, \tag{4.23}$$

for $j = 0, 1, \cdots, M-1$ and $i \in ROI_k$.

The above construction can also be viewed as each sensor i using a threshold η_i^k for quantization (as described in (3.3)). Let each region R_j^k correspond to a location θ_j^k for $j = 0, 1, \cdots, M-1$, which in our case is the center of the region R_j^k. Each sensor i decides on a '1' if and only if the target lies in the region $R_{r^k(i)}^k$. Every sensor i, therefore, performs a binary hypothesis test described as follows:

$$\begin{aligned} H_1 :& \quad \theta^k \in R^k_{r^k(i)} \\ H_0 :& \quad \theta^k \notin R^k_{r^k(i)}. \end{aligned} \tag{4.24}$$

If d_{i,θ^k_j} represents the Euclidean distance between the ith sensor and θ^k_j for $i = 1, 2, \cdots, N$ and $j = 0, 1, \cdots, M-1$, then $r^k(i) = \arg\min_l d_{i,\theta^k_l}$. Therefore, the condition $\theta^k \in R^k_{r^k(i)}$ can be abstracted as a threshold η^k_i on the local sensor signal amplitude given by

$$\eta^k_i = \frac{\sqrt{P_0}}{d_{i,\theta^k_{r^k(i)}}}. \tag{4.25}$$

This ensures that if the signal amplitude at the ith sensor is above the threshold η^k_i, then θ^k lies in region $R^k_{r^k(i)}$ leading to minimum distance decoding.

4.4.2 Performance Analysis

In this subsection, the performance of the proposed scheme is analyzed. An analytically tractable metric to analyze the performance of the proposed scheme is the probability of detection of the target region. It is an important metric when the final goal of the target localization task is to find the approximate region or neighborhood where the target lies rather than the true location itself. Since the final ROI could be one of the M regions, a metric of interest is the probability of 'zooming' into the correct region. In other words, it is the probability that the true location and the estimated location lie in the same region.

The final region of the estimated target location is the same as the true target location, if and only if we 'zoom' into the correct region at every iteration of the proposed scheme. If P^k_d denotes the detection probability at the $(k+1)$th iteration, the overall detection probability is given by

$$P_D = \prod_{k=0}^{k^{stop}} P^k_d. \tag{4.26}$$

Exact Analysis

Let us consider the $(k{+}1)$th iteration and define the received vector at the FC as $\boldsymbol{u}^k = [u_1^k, u_2^k, \cdots, u_{N_k}^k]$, where N_k are the number of local sensors reporting their data to FC at $(k+1)$th iteration. Let $\mathcal{D}_j^k$ be the decision region of jth hypothesis defined as follows:

$$\mathcal{D}_j^k = \{\boldsymbol{u}^k | d_H(\boldsymbol{u}^k, \boldsymbol{c}_{j+1}^k) \leq d_H(\boldsymbol{u}^k, \boldsymbol{c}_{l+1}^k) \text{ for } 0 \leq l \leq M-1\},$$

where $d_H(\cdot,\cdot)$ is the Hamming distance between two vectors, and $\boldsymbol{c}_{j+1}^k$ is the codeword corresponding to hypothesis j in code matrix C^k. Then define the reward $r_{\boldsymbol{u}^k}^{j,k}$ associated with the hypothesis j as

$$r_{\boldsymbol{u}^k}^{j,k} = \begin{cases} \frac{1}{q_{\boldsymbol{u}^k}} & \text{when } \boldsymbol{u}^k \in \mathcal{D}_j^k \\ 0 & \text{otherwise} \end{cases}, \tag{4.27}$$

where $q_{\boldsymbol{u}^k}$ is the number of decision regions to whom $\boldsymbol{u}^k$ belongs to. Note that $q_{\boldsymbol{u}^k}$ can be greater than one when there is a tie at the FC. Since the tie-breaking rule is to choose one of them randomly, the reward is given by (4.27). According to (4.27), the detection probability at the $(k{+}1)$th iteration is given by

$$\begin{aligned} P_d^k &= \sum_{j=0}^{M-1} P(H_j^k) \sum_{\boldsymbol{u}^k \in \{0,1\}^{N_k}} P(\boldsymbol{u}^k | H_j^k) r_{\boldsymbol{u}^k}^{j,k} \\ &= \frac{1}{M} \sum_{j=0}^{M-1} \sum_{\boldsymbol{u}^k \in \mathcal{D}_j^k} \left(\prod_{i=1}^{N_k} P(u_i^k | H_j^k) \right) \frac{1}{q_{\boldsymbol{u}^k}}, \end{aligned} \tag{4.28}$$

where $P(u_i^k|H_j^k)$ denotes the probability that the sensor i sends the bit $u_i^k \in \{0,1\}$, $i = 1, 2, \cdots, N_k$, when the true target is in the region R_j^k corresponding to H_j^k at the $(k+1)$th iteration.

From the system model described before,

$$P(u_i^k = 1 | H_j^k) = E_{\theta | H_j^k}\left[P(u_i^k = 1 | \theta, H_j^k) \right]. \tag{4.29}$$

Since (4.29) is complicated, it can be approximated using θ_j^k which is the center of the region R_j^k. (4.29) now simplifies to

$$P(u_i^k = 1|H_j^k) \approx Q\left(\frac{\eta_i^k - a_{ij}^k}{\sigma}\right), \tag{4.30}$$

where η_i^k is the threshold used by the ith sensor at kth iteration, σ^2 is the noise variance, a_{ij}^k is the signal amplitude received at the ith sensor when the target is at θ_j^k and $Q(x)$ is the complementary cumulative distribution function of standard Gaussian and is given by

$$Q(x) = \frac{1}{\sqrt{2\pi}} \int_x^\infty e^{(-t^2/2)} dt. \tag{4.31}$$

Using (4.26), the probability of detection of the target region can be found as the product of detection probabilities at every iteration k. It is clear from the derived expressions that the exact analysis of the detection probability is complicated and, therefore, some analytical bounds are derived on the performance of the proposed scheme.

Performance Bounds

In this section, performance bounds on the proposed coding based localization scheme are presented. For the analysis, the lemmas in [170] will be used, which are stated here for the sake of completeness.

Lemma 4.4.1 ([170]). *Let* $\{Z_j\}_{j=1}^\infty$ *be independent antipodal random variables with* $Pr[Z_j = 1] = q_j$ *and* $Pr[Z_j = -1] = 1 - q_j$. *If* $\lambda_m \triangleq E[Z_1 + \cdots + Z_m]/m < 0$, *then*

$$Pr\{Z_1 + \cdots + Z_m \geq 0\} \leq (1 - \lambda_m^2)^{m/2}. \tag{4.32}$$

Using this lemma, the performance bounds on our proposed scheme are presented.

Lemma 4.4.2. *Let* $\theta \in R_j^k$ *be the fixed target location. Let* $P_e^k(\theta)$ *be the error probability of detection of the target region given* θ *at the* $(k+1)$*th iteration. For the received vector of* $N_k = N/M^k$ *observations at the* $(k+1)$*th iteration,* $\boldsymbol{u}^k = [u_1^k, \cdots, u_{N_k}^k]$, *assume that for every* $0 \leq$

$j, l \leq M-1$ and $l \neq j$,

$$\sum_{i \in S_j^k \cup S_l^k} q_{i,j}^k < \frac{N_k}{M} = \frac{N}{M^{k+1}}, \tag{4.33}$$

where $q_{i,j}^k = P\{z_{i,j}^k = 1|\theta\}$, $z_{i,j}^k = 2(u_i^k \oplus c_{(j+1)i}^k) - 1$, and $C^k = \{c_{(j+1)i}^k\}$ is the code matrix used at the $(k+1)$th iteration. Then

$$\begin{aligned} P_e^k(\theta) &\leq \sum_{0 \leq l \leq M-1, l \neq j} \left(1 - \frac{\left(\sum_{i \in S_j^k \cup S_l^k} (2q_{i,j}^k - 1)\right)^2}{d_{m,k}^2}\right)^{d_{m,k}/2} && (4.34) \\ &\leq (M-1)\left(1 - \left(\lambda_{j,max}^k(\theta)\right)^2\right)^{d_{m,k}/2}, && (4.35) \end{aligned}$$

where $d_{m,k}$ is the minimum Hamming distance of the code matrix C^k given by $d_{m,k} = \frac{2N}{M^{k+1}}$ due to the structure of our code matrix and

$$\lambda_{j,max}^k(\theta) \triangleq \max_{0 \leq l \leq M-1, l \neq j} \frac{1}{d_{m,k}} \sum_{i \in S_j^k \cup S_l^k} (2q_{i,j}^k - 1). \tag{4.36}$$

Proof. The proof is provided in Appendix A.3. □

The probabilities $q_{i,j}^k = P\{u_i^k \neq c_{(j+1)i}^k|\theta\}$ can be easily computed as below. For $0 \leq j \leq M-1$ and $1 \leq i \leq N_k$, if $i \in S_j^k$,

$$\begin{aligned} q_{i,j}^k &= P\{u_i^k = 0|\theta\} \\ &= 1 - Q\left(\frac{(\eta_i^k - a_i)}{\sigma}\right), && (4.37) \end{aligned}$$

where η_i^k is the threshold used by the ith sensor at $(k+1)$th iteration, σ^2 is the noise variance, a_i is the amplitude received at the ith sensor given by (3.1) when the target is at θ. If $i \notin S_j^k$, $q_{i,j}^k = 1 - P\{u_i^k = 0|\theta\}$.

Before we present our main theorem, for ease of analysis, we give an assumption that will be used in the theorem. Note that, our proposed scheme can still be applied to those WSNs where the assumption does not hold.

Assumption 4.4.3. *For any target location* $\theta \in R_j^k$ *and any* $0 \leq k \leq k^{stop}$*, there exists a bijection function* f *from* S_j^k *to* S_l^k*, where* $0 \leq l \leq M-1$ *and* $l \neq j$*, such that*

$$f(i_j) = i_l,$$

$$\eta_{i_j}^k = \eta_{i_l}^k,$$

and

$$d_{i_j} < d_{i_l},$$

where $i_j \in S_j^k$*,* $i_l \in S_l^k$*, and* d_{i_j} *(*d_{i_l}*) is the distance between* θ *and sensor* i_j *(*i_l*).*

One example of WSNs that satisfies this assumption is given in Fig. 4.9. For every sensor $i_j \in S_j^k$, due to the symmetric region splitting, there exists a corresponding sensor $i_l \in S_l^k$ which is symmetrically located as described in the following: Join the centers of the two regions and draw a perpendicular bisector to this line as shown in Fig. 4.9. The sensor $i_l \in S_l^k$ is the sensor located symmetrically to sensor i_j on the other side of the line L. These are the sensors for which the thresholds are the same. In other words, due to the symmetric placement of the sensors, $\eta_{i_j}^k = \eta_{i_l}^k$. Clearly, when $\theta \in R_j^k$, $d_{i_j} < d_{i_l}$.

Theorem 4.4.4. *Let* P_D *be the probability of detection of the target region given by* (4.26)*, where* P_d^k *is the detection probability at the* $(k+1)$*th iteration. Under Assumption 4.4.3,*

$$P_d^k \geq 1 - (M-1)\left(1 - (\lambda_{max}^k)^2\right)^{d_{m,k}/2}, \tag{4.38}$$

where

$$\lambda_{max}^k \triangleq \max_{0 \leq j \leq M-1} \lambda_{j,max}^k$$

and

$$\lambda_{j,max}^k \triangleq \max_{\theta \in R_j^k} \lambda_{j,max}^k(\theta).$$

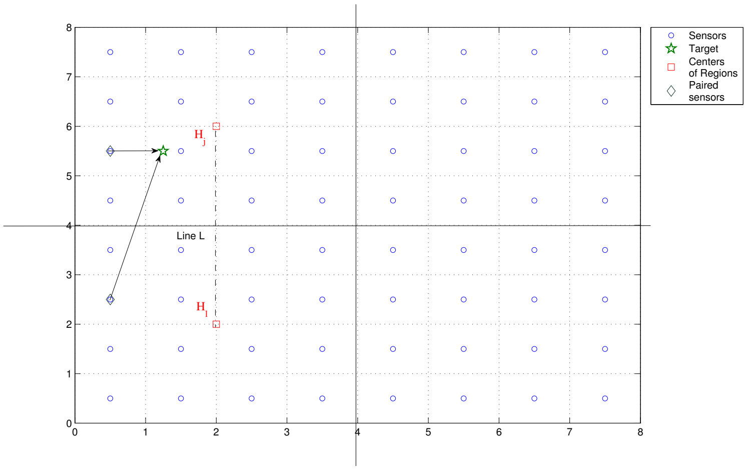


Fig. 4.9: ROI with an example set of paired sensors

Proof. The proof is provided in Appendix A.4. □

Next the asymptotic performance of the scheme is analyzed, i.e., P_D is examined for the case when N approaches infinity.

Theorem 4.4.5. *Under Assumption* (4.4.3), $\lim_{N\to\infty} P_D = 1$.

Proof. Note that

$$\lambda_{j,\max}^k = \max_{0\le l\le M-1, l\ne j} \frac{1}{d_{m,k}} \sum_{i\in S_j^k \cup S_l^k} (2q_{i,j}^k - 1) > \frac{M^{k+1}}{2N} \sum_{i\in S_j^k \cup S_l^k} (-1) = -1$$

for all $0 \le j \le M-1$ since not all $q_{i,j}^k = 0$. Hence, by definition, $\lambda_{\max}^k$ is also greater than -1. Since $-1 < \lambda_{\max}^k < 0$, $0 < 1-(\lambda_{\max}^k)^2 < 1$. Under the assumption that the number of iterations are finite, for a fixed number of regions M, the performance of the proposed scheme can be analyzed under asymptotic regime. Under this assumption, $d_{m,k} = \frac{2N}{M^{k+1}}$ grows linearly with the number of sensors N for $0 \leq k \leq k^{stop}$. Then

$$\begin{aligned}
\lim_{N\to\infty} P_D &= \lim_{N\to\infty} \prod_{k=0}^{k^{stop}} P_d^k \\
&\geq \prod_{k=0}^{k^{stop}} \lim_{N\to\infty} \left[(1-(M-1)(1-(\lambda_{\max}^k)^2)^{d_{m,k}/2}\right] \\
&= \prod_{k=0}^{k^{stop}} (1-(M-1) \lim_{N\to\infty} \left[(1-(\lambda_{\max}^k)^2)^{d_{m,k}/2}\right] \\
&= \prod_{k=0}^{k^{stop}} [1-(M-1)0] \\
&= \prod_{k=0}^{k^{stop}} 1 = 1.
\end{aligned}$$

Hence, the overall detection probability becomes ‘1’ as the number of sensors N goes to infinity. This shows that the proposed scheme asymptotically attains perfect region detection probability irrespective of the value of finite noise variance. $\square$

4.4.3 Numerical Results

Some numerical results are now presented which justify the analytical results presented in the previous subsection and provide some insights. In the previous subsection, it was observed that the performance of the basic coding scheme quantified by the probability of region detection asymptotically approaches ‘1’ irrespective of the finite noise variance. Fig. 4.10 shows that the region detection probability increases as the number of sensors approaches infinity. Observe that for a fixed noise variance, the region detection probability increases with increase in the number of sensors. Also, for a fixed number of sensors, the region detection probability decreases with σ when the number of sensors are low. But when the number of sensors is large, the reduction in region detection probability with σ is negligible and as $N \to \infty$, the region detection probability converges to 1.

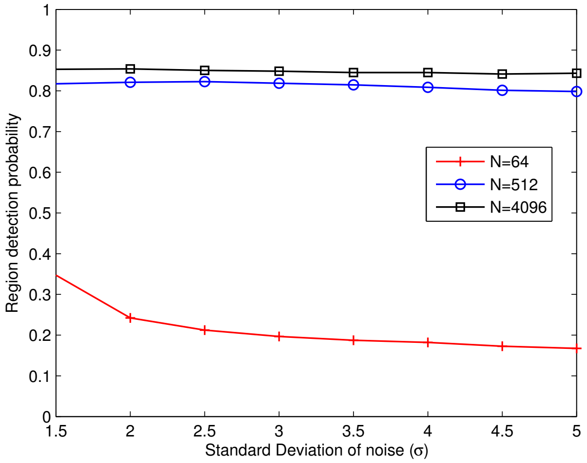


Fig. 4.10: Region detection probability versus the standard deviation of noise with varying number of sensors

4.5 Localization in the Presence of Byzantines

Now consider the case when there are Byzantines in the network. As discussed before, Byzantines are local sensors which send false information to the FC to deteriorate the network's performance. Assume the presence of $B = \alpha N$ number of Byzantines in the network. Here, the Byzantines are assumed to attack the network independently where the Byzantines flip their data with probability '1' before sending it to the FC.[2] In other words, the data sent by the ith sensor is given by:

$$u_i = \begin{cases} D_i & \text{if } i\text{th sensor is honest} \\ \bar{D}_i & \text{if } i\text{th sensor is Byzantine} \end{cases}. \tag{4.39}$$

For such a system, it has been shown in the previous chapter that the FC becomes blind to network's information for $\alpha \geq 0.5$. Therefore, for the remainder of this chapter, the system is analyzed when $\alpha < 0.5$. For the basic coding scheme described in Sec. 4.4.1, each column in C^k contains only one '1' and every row of C^k contains exactly $\frac{N}{M^{k+1}}$ '1's. Therefore, the minimum Hamming distance of C^k is $\frac{2N}{M^{k+1}}$ and, at the $(k+1)$th iteration, it can tolerate a total of at most

---

[2] It has been shown in the previous chapter that the optimal independent attack strategy for the Byzantines is to flip their data with probability '1'.

$\frac{N}{M^{k+1}} - 1$ faults (data falsification attacks) due to the presence of Byzantines in the network. This value is not very high and the basic scheme is now extended to a scheme which can handle more Byzantine faults.

4.5.1 Exclusion Method with Weighted Average

As shown above, the scheme proposed in Sec. 4.4.1 has a Byzantine fault tolerance capability which is not very high. The performance can be improved by using an exclusion method for decoding where the best two regions are kept for the next iteration and using weighted average to estimate the target location at the final step. This scheme builds on the basic coding scheme proposed in Sec. 4.4.1 with the following modifications:

- Since after every iteration two regions are kept, the code matrix after the kth iteration is of size $M \times \frac{2^k N}{M^k}$ and the number of iterations needed to stop the localization task needs to satisfy $k^{stop} < \log_{M/2} N$.
- At the final step, instead of taking an average of the sensor locations of the sensors present in the ROI at the final step, we take a weighted average of the sensor locations where the weights are the 1-bit decisions sent by these sensors. Since, a decision $u_i = 1$ implies that the target is closer to sensor i, a weighted average ensures that the average is taken only over the sensors for which the target is reported to be close.

Therefore, the target location estimate is given by

$$\hat{\theta}_x = \frac{\sum_{i \in ROI_{k^{stop}}} u_i x_i}{\sum_{i \in ROI_{k^{stop}}} u_i} \tag{4.40}$$

$$\text{and} \quad \hat{\theta}_y = \frac{\sum_{i \in ROI_{k^{stop}}} u_i y_i}{\sum_{i \in ROI_{k^{stop}}} u_i}. \tag{4.41}$$

The exclusion method results in a better performance compared to the basic coding scheme since it keeps the two best regions after every iteration. This observation is also evident in the numerical results presented in Sec. 4.5.3.

4.5.2 Performance Analysis

Byzantine Fault Tolerance Capability

When the exclusion method based scheme described in Sec. 4.5.1 is used, since the two best regions are considered after every iteration, the fault tolerance performance improves and a total of at most $\frac{2^{k+1}N}{M^{k+1}}-1$ faults can be tolerated. This improvement in the fault tolerance capability can be observed in the simulation results presented in Sec. 4.5.3.

Proposition 4.5.1. *The maximum fraction of Byzantines that can be handled at the* $(k+1)$*th iteration by the proposed exclusion method based coding scheme is limited by* $\alpha_f^k=\frac{2}{M}-\frac{M^k}{2^kN}$.

Proof. The proof is straight forward and follows from the fact that the error correcting capability of the code matrix C^k at $(k+1)$th iteration is at most $\frac{2^{k+1}N}{M^{k+1}}-1$. Since there are $\frac{2^kN}{M^k}$ sensors present during this iteration, the fraction of Byzantine sensors that can be handled is given by $\alpha_f^k=\frac{2}{M}-\frac{M^k}{2^kN}$. □

The performance bounds on the basic coding scheme presented in Sec. 4.4.2 can be extended to the exclusion based coding scheme presented in Sec. 4.5.1. When there are Byzantines in the network, the probabilities $q_{i,j}^k$ of (4.37) become

$$q_{i,j}^k \;=\; 1-\left[(1-\alpha)Q\left(\frac{(\eta_i^k-a_i)}{\sigma}\right)+\alpha\left(1-Q\left(\frac{(\eta_i^k-a_i)}{\sigma}\right)\right)\right]. \tag{4.42}$$

It was shown in Sec. 4.4.2 that the detection probability at every iteration approaches '1' as the number of sensors N goes to infinity. However, this result only holds when the condition in (4.33) is satisfied. Notice that, in the presence of Byzantines,

$$q_{i,j}^k=\begin{cases}(1-\alpha)\left(1-Q\left(\frac{(\eta_i^k-a_i)}{\sigma}\right)\right)+\alpha Q\left(\frac{(\eta_i^k-a_i)}{\sigma}\right), & \text{for } i\in S_j^k\\ (1-\alpha)Q\left(\frac{(\eta_i^k-a_i)}{\sigma}\right)+\alpha\left(1-Q\left(\frac{(\eta_i^k-a_i)}{\sigma}\right)\right), & \text{for } i\in S_l^k\end{cases}, \tag{4.43}$$

which can be simplified as

$$q_{i,j}^k = \begin{cases} (1-\alpha) - (1-2\alpha)Q\left(\frac{(\eta_i^k - a_i)}{\sigma}\right), & \text{for } i \in S_j^k \\ \alpha + (1-2\alpha)Q\left(\frac{(\eta_i^k - a_i)}{\sigma}\right), & \text{for } i \in S_l^k \end{cases}. \tag{4.44}$$

Now using the pairwise sum approach discussed in Sec. 4.4.2, (A.17) can be re-written as follows:

$$q_{i_j,j}^k + q_{i_l,j}^k \;=\; 1 - (1-2\alpha)\left[Q\left(\frac{(\eta - a_{i_j})}{\sigma}\right) - Q\left(\frac{(\eta - a_{i_l})}{\sigma}\right)\right], \tag{4.45}$$

which is an increasing function of α since $Q\left(\frac{(\eta - a_{i_j})}{\sigma}\right) > Q\left(\frac{(\eta - a_{i_l})}{\sigma}\right)$ for all finite σ as discussed before. Therefore, when $\alpha < 0.5$, the pairwise sum in (4.45) is strictly less than 1 and the condition (4.33) is satisfied. However, when $\alpha \geq 0.5$, $\sum_{i \in S_j^k \cup S_l^k} q_{i,j}^k \geq \frac{N_k}{M}$. Therefore, the condition fails when $\alpha \geq 0.5$. It has been shown in the previous chapter that the FC becomes blind to the local sensor's information when $\alpha \geq 0.5$. Next, we state the theorem when there are Byzantines in the network.

Theorem 4.5.2. *Let α be the fraction of Byzantines in the networks. Under Assumption* (4.4.3)*, when $\alpha < 0.5$, $\lim_{N \to \infty} P_D = 1$.*

Note that the performance bounds derived can be used for system design. Let us consider N sensors uniformly deployed in a square region. Let this region be split into M equal regions. From Prop. 4.5.1, we know that α_f^k is a function of M and N. Also, the detection probability equations and bounds derived in Sec. 4.4.2 are functions of M and N. Hence, for given fault tolerance capability and region detection probability requirements, one can find the corresponding number of sensors (N_{req}) to be used and the number of regions to be considered at each iteration (M_{req}). Some guidelines for system design of a network which adopts the proposed approach are presented in the following. Suppose that a system is to be designed that splits into $M = 4$ regions after every iteration. How should a system designer decide the number of sensors N in order to meet the target region detection probability and Byzantine fault tolerance capability requirements? Table 4.2 shows the performance of the system in terms of the target region detection probability and Byzantine fault tolerance capability with varying number of sensors found using the expressions derived in Prop. 4.5.1 and in Sec. 4.4.2.

Table 4.2: Target region detection probability and Byzantine fault tolerance capability with varying N ($M = 4$)

N	Target Region Detection probability	Byzantine fault tolerance capability
32	0.4253	0.4688
128	0.6817	0.4844
512	0.6994	0.4922

From Table 4.2, it can be observed that the performance improves with increasing number of sensors. However, as a system designer, one would like to minimize the number of sensors that need to be deployed while assuring a minimum performance guarantee. In this example, if one is interested in achieving a region detection probability of approximately 0.7 and a Byzantine fault tolerance capability close to 0.5, $N = 512$ sensors are sufficient.

4.5.3 Simulation Results

In this section, some simulation results are presented to evaluate the performance of the proposed schemes in the presence of Byzantine faults. The performance is analyzed using two performance metrics: mean square error (MSE) of the estimated location and probability of detection (P_D) of the target region. A network of $N = 512$ sensors are deployed in a regular 8×8 grid as shown in Fig. 4.8. Let α denote the fraction of Byzantines in the network that are randomly distributed over the network. The received signal amplitude at the local sensors is corrupted by AWGN noise with noise standard deviation $\sigma = 3$. The power at the reference distance is $P_0 = 200$. At every iteration, the ROI is split into $M = 4$ equal regions as shown in Fig. 4.8. We stop the iterations for the basic coding scheme after $k^{stop} = 2$ iterations. The number of sensors in the ROI at the final step are therefore, 32. In order to have a fair comparison, we stop the exclusion method after $k^{stop} = 4$ iterations, so that there are again 32 sensors in the ROI at the final step.

Fig. 4.11 shows the performance of the proposed schemes in terms of the MSE of the estimated target location when compared with the traditional maximum likelihood estimation [101]. The

MSE has been found by performing 1×10^3 Monte Carlo runs with the true target location randomly chosen in the 8×8 grid.

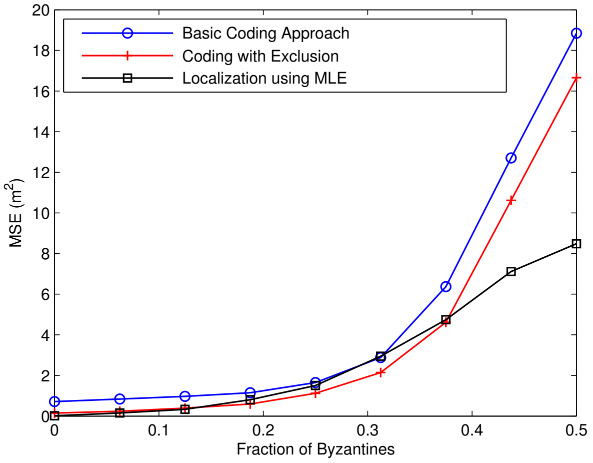


Fig. 4.11: MSE comparison of the three localization schemes: Basic coding scheme, Coding with exclusion, and ML-based localization

As can be seen from Fig. 4.11, the performance of the exclusion method based coding scheme is better than the basic coding scheme and outperforms the traditional MLE based scheme when $\alpha \leq 0.375$. When $\alpha > 0.375$, the traditional MLE based scheme has the best performance.

However, it is important to note that the proposed schemes provide a coarse estimate as against the traditional MLE based scheme which optimizes over the entire ROI. Also, the traditional scheme is computationally much more expensive than the proposed coding based schemes. In the simulations performed, the proposed schemes are around 150 times faster than the conventional scheme when the global optimization toolbox in MATLAB was used for the optimization in the ML based scheme. The computation time is very important in a scenario when the target is moving and a coarse location estimate is needed in a timely manner.

Fig. 4.12 shows the performance of the proposed schemes in terms of the detection probability of the target region. The detection probability has been found by performing 1×10^4 Monte Carlo runs with the true target randomly chosen in the ROI. Fig. 4.12 shows the reduction in the detection probability with an increase in α when more sensors are Byzantines sending false information to the FC.

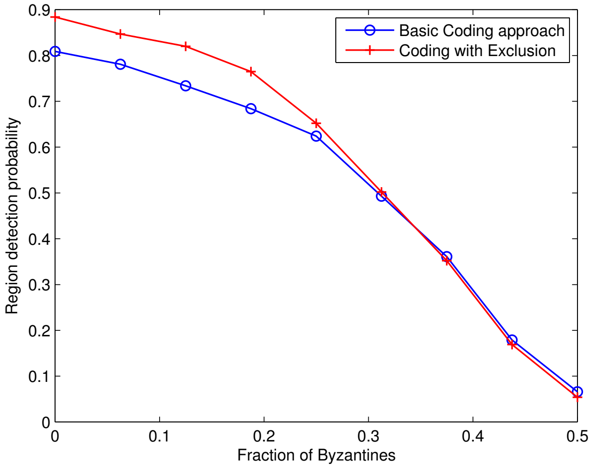


Fig. 4.12: Probability of detection of target region as a function of α

In order to analyze the effect of the number of sensors on the performance, simulations were performed by changing the number of sensors and keeping the number of iterations the same as before. According to Prop. 4.5.1, when $M = 4$, the proposed scheme can asymptotically handle up to 50% of the sensors being Byzantines. Figs. 4.13 and 4.14 show the effect of the number of sensors on MSE and detection probability of the target region respectively when the exclusion method based coding scheme is used. As can be seen from both figures (Figs. 4.13 and 4.14), the fault-tolerance capability of the proposed scheme improves with an increase in the number of sensors and approaches $\alpha_f^k = 0.5$ asymptotically.

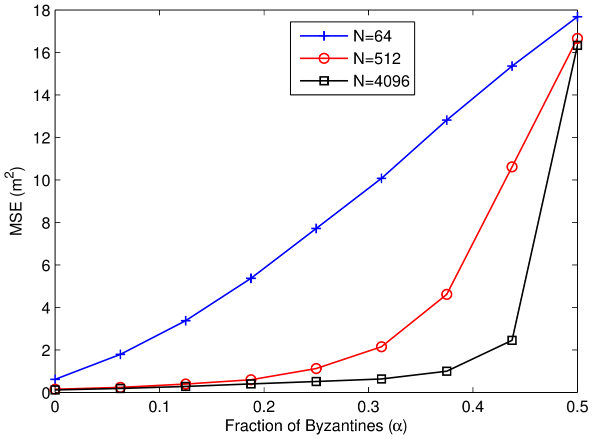


Fig. 4.13: MSE of the target location estimate with varying N

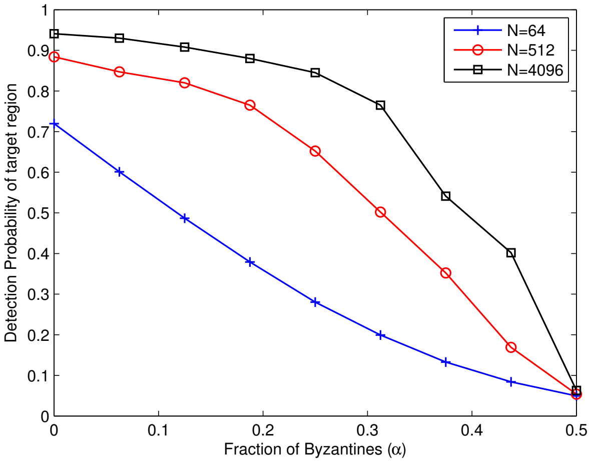


Fig. 4.14: Probability of detection of target region with varying N

4.6 Soft-Decision Decoding for Non-Ideal Channels

While Byzantines have been considered in the previous section, another practical cause of unreliability in the observations is the presence of imperfect channels. In this section, the scheme is extended to counter the effect of non-ideal channels on system performance. Besides the faults due to the Byzantines in the network, the presence of non-ideal channels further degrades the localization performance. To combat the channel effects, a soft-decision decoding rule is used at every iteration, instead of the minimum Hamming distance decoding rule.

4.6.1 Decoding Rule

At each iteration, the local sensors transmit their local decisions $\boldsymbol{u}^k$ which are possibly corrupted due to the presence of Byzantines. Let the received analog data at the FC be represented as $\boldsymbol{v}^k = [v_1^k, v_2^k, \cdots, v_{N_k}^k]$, where the received observations are related to the transmitted decisions as follows:

$$v_i^k = h_i^k(-1)^{u_i^k}\sqrt{E_b} + n_i^k, \qquad \forall i = \{1, \cdots, N_k\}, \tag{4.46}$$

where h_i^k is the fading channel coefficient, E_b is the energy per channel bit and n_i^k is the additive white Gaussian noise with variance σ_f^2. Here, the channel coefficients are assumed to be Rayleigh distributed with variance σ_h^2.

The FC is assumed to have no knowledge of the fraction of Byzantines α. Hence, instead of adopting the reliability measure given in (2.2), a simpler reliability measure ψ_i^k is used in our decoding rule that is not related to local decisions of sensors. It will be shown that this reliability measure captures the effect of imperfect channels reasonably well when there are Byzantines in the network. The reliability for each of the received bits is defined as follows:

$$\psi_i^k = \ln \frac{P(v_i^k | u_i^k = 0)}{P(v_i^k | u_i^k = 1)} \tag{4.47}$$

for $i = \{1, \cdots, N\}$. Here $P(v_i^k | u_i^k)$ can be obtained from the statistical model of the Rayleigh fading channel considered here. Define F-distance as

$$d_F(\boldsymbol{\psi}^k, \boldsymbol{c}_{j+1}^k) = \sum_{i=1}^{N_k} (\psi_i^k - (-1)^{c_{(j+1)i}^k})^2,$$

where $\boldsymbol{\psi}^k = [\psi_1^k, \cdots, \psi_{N_k}^k]$ and $\boldsymbol{c}_{j+1}^k$ is the jth row of the code matrix C^k. Then, the fusion rule is to decide the region R_j^k for which the F-distance between $\boldsymbol{\psi}^k$ and the row of C^k corresponding to R_j^k is minimized.

4.6.2 Performance Analysis

In this section, some bounds on the performance of the soft-decision decoding scheme are presented in terms of the detection probability. Without loss of generality, assume $E_b = 1$. As mentioned before in (4.26), the overall detection probability is the product of the probability of detection at each iteration, P_d^k. The following lemma is first presented without proof which is used to prove the theorem stated later in this section.

Lemma 4.6.1 ([165]). *Let* $\tilde{\psi}_i^k = \psi_i^k - E[\psi_i^k | \theta]$, *then*

$$E\left[(\tilde{\psi}_i^k)^2 | \theta\right] \leq \frac{8}{\sigma^4} \left\{ E[(h_i^k)^4] + E[(h_i^k)^2] \sigma_f^2 \right\}, \tag{4.48}$$

where σ^2 is the variance of the noise at the local sensors whose observations follow (3.2). For the Rayleigh fading channel considered here, both $E[(h_i^k)^4]$ and $E[(h_i^k)^2]$ are bounded and therefore, the LHS of (4.48) is also bounded.

Lemma 4.6.2. *Let $\theta \in R_j^k$ be the fixed target location. Let $P_{e,j}^k(\theta)$ be the error probability of detection of the target region given $\theta \in R_j^k$ at the $(k+1)$th iteration. For the reliability vector $\boldsymbol{\psi}^k = [\psi_1^k, \cdots, \psi_{N_k}^k]$ of the $N_k = N/M^k$ observations and code matrix C^k used at the $(k+1)$th iteration,*

$$P_{e,j}^k(\theta) \leq \sum_{0 \leq l \leq M-1, l \neq j} P\left\{ \sum_{i \in S_j^k \cup S_l^k} Z_i^{jl} \tilde{\psi}_i^k < - \sum_{i \in S_j^k \cup S_l^k} Z_i^{jl} E[\psi_i^k | \theta] \middle| \theta \right\}, \tag{4.49}$$

where $Z_i^{jl} = \frac{1}{2}((-1)^{c_{ji}^k} - (-1)^{c_{li}^k})$.

Proof.

$$\begin{aligned}
P_{e,j}^k(\theta) &= P\{\text{detected region} \neq R_j^k | \theta\} \\
&\leq P\left\{ d_F(\boldsymbol{\psi}^k, \boldsymbol{c}_{j+1}^k) \geq \min_{0 \leq l \leq M-1, l \neq j} d_F(\boldsymbol{\psi}^k, \boldsymbol{c}_{l+1}^k) | \theta \right\} \\
&\leq \sum_{0 \leq l \leq M-1, l \neq j} P\left\{ d_F(\boldsymbol{\psi}^k, \boldsymbol{c}_{j+1}^k) \geq d_F(\boldsymbol{\psi}^k, \boldsymbol{c}_{l+1}^k) | \theta \right\} \\
&= \sum_{0 \leq l \leq M-1, l \neq j} P\left\{ \sum_{i=1}^{N_k} (\psi_i^k - (-1)^{c_{(j+1)i}^k})^2 \geq (\psi_i^k - (-1)^{c_{(l+1)i}^k})^2 | \theta \right\} \\
&= \sum_{0 \leq l \leq M-1, l \neq j} P\left\{ \sum_{i \in S_j^k \cup S_l^k} Z_i^{jl} \psi_i^k < 0 \middle| \theta \right\} \\
&= \sum_{0 \leq l \leq M-1, l \neq j} P\left\{ \sum_{i \in S_j^k \cup S_l^k} Z_i^{jl} \tilde{\psi}_i^k < - \sum_{i \in S_j^k \cup S_l^k} Z_i^{jl} E[\psi_i^k | \theta] \middle| \theta \right\}.
\end{aligned} \tag{4.50}$$

□

Let $\sigma_{\tilde{\psi}}^2(\theta) = \sum_{i \in S_j^k \cup S_l^k} E\left[(Z_i^{jl} \tilde{\psi}_i^k)^2 | \theta \right] = \sum_{i \in S_j^k \cup S_l^k} E\left[(\tilde{\psi}_i^k)^2 | \theta \right]$, then the above result can be re-written as

$$P_{e,j}^k(\theta) \leq \sum_{0 \leq l \leq M-1, l \neq j} P\left\{ \frac{1}{\sigma_{\tilde{\psi}}(\theta)} \sum_{i \in S_j^k \cup S_l^k} Z_i^{jl} \tilde{\psi}_i^k < -\frac{1}{\sigma_{\tilde{\psi}}(\theta)} \sum_{i \in S_j^k \cup S_l^k} Z_i^{jl} E[\psi_i^k | \theta] \middle| \theta \right\}. \tag{4.51}$$

Under the assumption that for a fixed M, $\frac{N}{M^{k+1}} \to \infty$ as $N \to \infty$ for $k = 0, \cdots, k^{stop}$, we have the following result for asymptotic performance of the proposed soft-decision rule decoding based scheme.

Theorem 4.6.3. *Under Assumption* (4.4.3)*, when* $\alpha < 0.5$*,*

$$\lim_{N \to \infty} P_D = 1.$$

Proof. The proof is provided in Appendix A.5. □

Note that the detection probability of the proposed scheme can approach '1' even for extremely bad channels with very low channel capacity. This is true because, for fixed M, when N approaches infinity, the code rate of the code matrix approaches zero. Hence, even for extremely bad channels, the code rate is still less than the channel capacity.

4.6.3 Numerical Results

In this section, some numerical results are presented which show the improvement in the system performance when soft-decision decoding rule is used instead of the hard-decision decoding rule in the presence of Byzantines and non-ideal channels. As defined before, α represents the fraction of Byzantines and evaluate the performance of the basic coding approach with soft-decision decoding at the FC. Consider the scenario with following system parameters: $N = 512$, $M = 4$, $A = 8^2 = 64$ sq. units, $P_0 = 200$, $\sigma = 3$, $E_b = 1$, $\sigma_f = 3$ and $E[(h_i^k)^2] = 1$ which corresponds to $\sigma_h^2 = 1 - \frac{\pi}{4}$. The basic coding approach is stopped after $k^{stop} = 2$ iterations. Note that in the presence of non-ideal channels, α_{blind} is less than 0.5 since the non-ideal channels add to the errors at the FC. The number of Byzantine faults which the network can handle reduces and is now less than 0.5. In the simulations, the performance of the schemes deteriorates significantly when $\alpha \to 0.4$ (as opposed to 0.5 observed before) and therefore, the results are only plotted for the case when $\alpha \leq 0.4$

Fig. 4.15 shows the reduction in mean square error when the soft-decision decoding rule is used instead of the hard-decision decoding rule. Similarly, Fig. 4.16 shows the improvement in target region detection probability when using the soft-decision decoding rule. The plots are for 5×10^3 Monte-Carlo simulations.

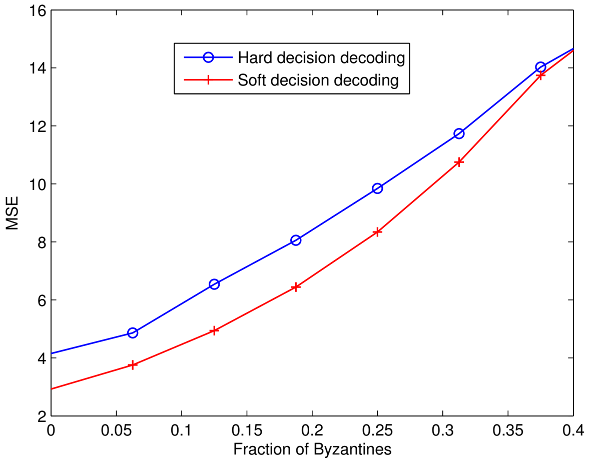


Fig. 4.15: MSE comparison of the basic coding scheme using soft- and hard- decision decoding

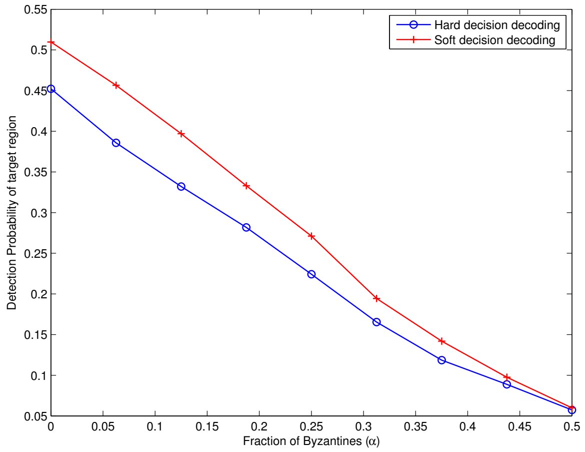


Fig. 4.16: Probability of detection of target region comparison of the basic coding scheme using soft- and hard- decision decoding

As the figures suggest, the performance deteriorates in the presence of non-ideal channels.

Also, the performance worsens with an increase in the number of Byzantines. The performance can be improved by using the exclusion method based coding approach as discussed in Sec. 4.5 in which two regions are stored after every iteration. Figs. 4.17 and 4.18 show this improved performance as compared to the basic coding approach. Note that the exclusion method based coding approach also follows the same trend as the basic coding approach with soft-decision decoding performing better than hard-decision decoding.

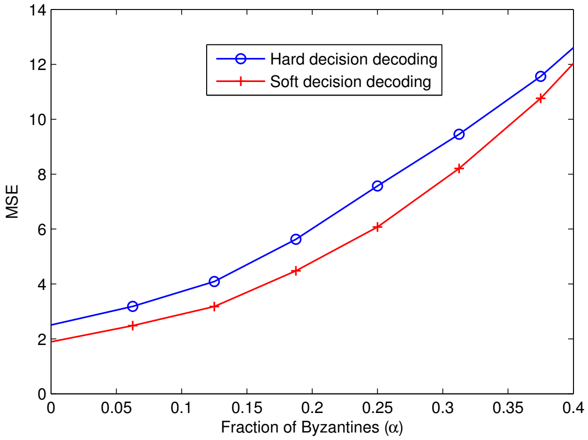


Fig. 4.17: MSE comparison of the exclusion coding scheme using soft- and hard- decision decoding

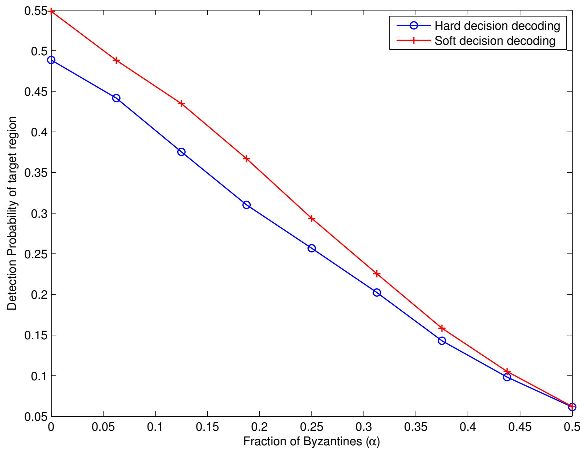


Fig. 4.18: Probability of detection of target region comparison of the exclusion coding scheme using soft- and hard- decision decoding

In the theoretical analysis, it was shown that the probability of region detection asymptotically approaches '1' irrespective of the finite noise variance. Fig. 4.19 presents this result that the region detection probability increases as the number of sensors approach infinity. Observe that for a fixed noise variance, the region detection probability increases with an increase in the number of sensors and approaches '1' as $N \to \infty$. However, as σ_f increases, the convergence rate decreases. For example, when $\sigma_f = 1.5$, $N = 4096$ is large enough to have a P_D value close to 0.85. However, for $\sigma_f = 4$, $N = 4096$ results in $P_D = 0.65$ which is not very large. It is expected that $P_D \to 1$ only for much larger number of sensors for $\sigma_f = 4$ and therefore, the convergence rate is less compared to when $\sigma_f = 1.5$.

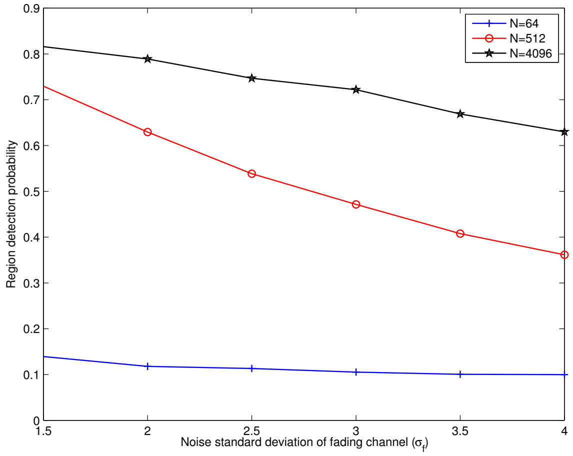


Fig. 4.19: Probability of detection of target region of the exclusive coding scheme using soft-decision decoding with varying number of sensors

4.7 Discussion

In this chapter, the design of reliable localization and tracking in sensor networks has been explored when the network consists of some malicious sensors referred to as Byzantines. Based on the results on the nature of Byzantines derived in Chapter 3, three schemes have been proposed. Each of these schemes are of increasing complexity and propose more changes in the system mechanisms. The first scheme deals with identification of the Byzantines by observing the data over time. It was shown that the proposed scheme works well and identifies most of the Byzantines. In order to improve the performance further, the second scheme moves from the traditional static identical thresholds at the local sensors to dynamic non-identical thresholds. By doing so, one can improve the system performance while also making the Byzantines ineffective in their attack strategy. Both these schemes deal with maximum likelihood type (MMSE) optimal estimators at the FC. The third scheme proposed a sub-optimal but simple and effective estimator at the FC that is based on coding scheme. This scheme is shown to be computationally more efficient while also making it robust to Byzantine attacks due to the use of error-correcting codes. Asymptotic optimality of the scheme is proved using large-deviation techniques.

CHAPTER 5

ESTIMATION IN HUMAN NETWORKS: UNRELIABLE LOCAL AGENTS

5.1 Introduction

In the previous chapters, inference in sensor networks has been considered. The effect of unreliable sensors on inference in the network was analyzed in Chapter 3 and schemes to mitigate their effect and ensure reliable inference at the FC were discussed in Chapter 4. In the following chapters (Chapters 5–7), human networks are considered and a similar approach is followed. In this chapter, the effect of unreliable local humans is considered. When all the agents in the network are humans, we have the human networks such as team decision making systems typically seen in large organizations such as firms. For example, consider the problem faced by the chief executive officer (CEO) of a firm with a large portfolio of projects that each have an underlying probability of success. Each of the subordinates will have noisy beliefs about the risks facing the projects. The CEO has cognitive constraints that limit the information rate he/she can receive from the subordinates, requiring subordinates to partition risks into quantal grades like A, B, C, and D, before conveying them. Such quantized grading is typical in businesses with complex information technology projects [115]. Upon receiving information from the subordinate agents, the CEO estimates the underlying success probability to minimize mean squared error (Brier score [110]) in estimating risk before taking action.

This is, of course, a version of the classical problem in multiterminal source coding called the CEO problem, but for a source that has not been considered in the literature. The source considered here is a non-regular source distribution and is defined as follows: an i.i.d. source sequence $X(t)$, which follows a probability density function (pdf) $f_X(x)$ with finite support $\mathcal{X}$, is considered to be non-regular if

$$\frac{\partial f_X(x)}{\partial x} \text{ or } \frac{\partial^2 f_X(x)}{\partial x^2}$$

either does not exist or is not absolutely integrable. The CEO problem was first introduced by Berger, Zhang, and Viswanathan [17], where they considered the source of interest to be a discrete data source sequence and studied the asymptotic behavior of the minimal error probability in the limit as the number of agents and the sum rate tend to infinity. It was later extended to the case when the source sequence of interest is continuous and distributed as Gaussian and a quadratic distortion measure is used as the performance measure [162]. Oohama studied the sum rate distortion function of the quadratic Gaussian CEO problem and determined the complete solution to the problem [102]; the full rate-distortion region was then found independently by Oohama [103] and Prabhakaran, et al. [109]. Several extensions to this problem have been studied [35, 36, 52, 133, 138, 164, 169]. However, most of these extensions continue to deal with the quadratic Gaussian setting. Viswanath formulated a similar multiterminal Gaussian source coding problem and characterized the sum rate distortion function for a class of quadratic distortion metrics [161]. In [35, 138], the authors consider the vector Gaussian case and study the sum rate for the vector Gaussian CEO problem. The related problem of determining the rate region for the quadratic Gaussian two-encoder source coding problem was solved by Wagner, Tavildar, and Viswanath [164]. Chen, et al. determined bounds on the rate region for the CEO problem with general source distributions [36]. Eswaran and Gastpar considered the CEO problem where the source is non-Gaussian but observations are still made through an additive white Gaussian noise (AWGN) channel [52]. The above problem of belief sharing in organizations is also closely connected to studies of communicating probability values [60, 79, 118, 148, 149]. Contrary to typical work in distributed source coding for inference [57] that is concerned with compressing agents' measurements, optimal quantization of prior probabilities for Bayesian hypothesis testing was studied in [148]. This was extended to the case of collaborative decision making in parallel fusion settings [118], very much like the CEO problem herein but in non-asymptotic regimes. Such a problem arises in several statistical signal processing, economics, and political science settings such as human affairs, where juries or committees need to have a common preference between two given alternatives.

Although, the problem is motivated by the perspective of inference by humans, such situations may be faced even in the case of sensors. In sensor network settings where sensors see a phenomenon through Gaussian noise but produce censored data due to hardware limitations of the measuring device, observations might follow truncated Gaussian with bounded support. Censored sensor data renders celebrated information-theoretic results for Gaussian observations invalid. The results in this chapter hold under such scenarios when the sensor observations follow truncated Gaussian distribution, a non-regular distribution.

The remainder of the chapter is organized as follows: In Sec. 5.2, the mathematical formulation of the non-regular CEO problem is described and the main result is stated. The achievability is proved in Sec. 5.3 using a layered architecture with scalar quantization, distributed entropy coding, and midrange estimation. The converse is proved in Sec. 5.4 using the Bayesian Chazan-Zakai-Ziv bound. Concluding remarks are presented in Sec. 5.5.

5.2 Non-Regular CEO Problem

Consider an i.i.d. source sequence of interest $\{X(t)\}_{t=1}^{\infty}$ drawn from a non-regular probability density function $f_X(x)$ with finite support $\mathcal{X}$. Without loss of generality, let the source be supported on $[0, 1]$.[1] Several agents (L) make imperfect conditionally independent assessments of $\{X(t)\}_{t=1}^{\infty}$,

---

[1] Note that an extension to a general finite support is straightforward.

to obtain noisy versions $\{Y_i(t)\}_{t=1}^{\infty}$ for $i = 1, \ldots, L$. The relationship between $X(t)$ and $Y_i(t)$ is governed by a conditional probability density function $W_\alpha(y_i|x)$ for all agents, where α is the coupling parameter. This coupling parameter represents the strength of the dependence between the source $X(t)$ and the observations $Y_i(t)$. The agents separately compress their observations. The CEO is interested in estimating $X(t)$ such that the mean squared error (MSE) between the n-block source $X^n = [X(1), \ldots, X(n)]$ and its estimate $\hat{X}^n = [\hat{X}(1), \ldots, \hat{X}(n)]$ is minimized.

Let the source code $\mathcal{C}_i^n$ of rate $R_i^n = (1/n) \log |\mathcal{C}_i^n|$ represent the coding scheme used by agent i to encode a block of length n of observed data $\{y_i(t)\}_{t=1}^{\infty}$. The CEO's estimate is given as $\hat{X}^n = \phi_L^n(C_1^n, \ldots, C_L^n)$ where $\phi_L^n : C_1^n \times \cdots \times C_L^n$ is the CEO's mapping. A specific achievability scheme for an example system, Fig. 5.1 shown in the sequel, illustrates the basic system structure.

We are interested in the tradeoff between the sum rate $R = \sum_{i=1}^{L} R_i^n$ and the MSE at the CEO, $D^n(X^n, \hat{X}^n)$, defined as:

$$D^n(X^n, \hat{X}^n) \triangleq \frac{1}{n} \sum_{t=1}^{n} (X(t) - \hat{X}(t))^2. \tag{5.1}$$

For a fixed set of codes, the MSE corresponding to the best estimator at the CEO is given by:

$$D^n(C_1^n, \ldots, C_L^n) \triangleq \min_{\phi_L^n} D^n(X^n, \phi_L^n(C_1^n, \ldots, C_L^n)).$$

Also, define the following quantities:

$$D^n(L, R) \triangleq \min_{\{C_i^n\} : \sum_{i=1}^{L} R_i^n \leq R} D^n(C_1^n, \ldots, C_L^n), \tag{5.2}$$

$$D(L, R) \triangleq \lim_{n \to \infty} D^n(L, R), \tag{5.3}$$

and

$$D(R) \triangleq \lim_{L \to \infty} D(L, R). \tag{5.4}$$

To understand the tradeoff between sum rate and distortion, the following quantity is studied:

$$\beta(\alpha) \triangleq \lim_{R\to\infty} R^2 D(R).$$

Let X be the generic random variable representing the source and Y_i represent the generic random variable representing agent i's observation where X and Y_i are related through the conditional pdf $W_\alpha(y_i|x)$. The focus is on observation channels which satisfy the following property, when a forward test channel with an output auxiliary random variable U, is used.

Property 5.2.1. *For a given observation channel $W_\alpha(y|x)$ between X and Y, there exists a random variable U such that: X, Y, and U form a Markov chain, $X \to Y \to U$, and the conditional distribution of U given X, $f_{U|X}(u|x)$, has bounded support: $u \in [a(x), b(x)]$, where $(a+b)(x)$ is invertible and the inverse function $l(\cdot) := (a+b)^{-1}(\cdot)$ is Lipschitz continuous with Lipschitz constant $K > 0$, and further does not vanish at its end points: $\lim_{u\to a(x)\ or\ b(x)} f_{U|X}(u|x) > 0$.*

Let the set $\mathcal{S}(W)$ denote the set of random variables U which satisfy the above property for a given observation channel W. Explicit examples of channels satisfying this property are provided later in Sec. 5.2.1.

The main result is now stated here; achievability and converse proofs are developed in the sequel.

Theorem 5.2.2. *When conditional density W_α satisfies Property 5.2.1, the following relations hold:*

$$\beta(\alpha) \leq \frac{2K^2}{\delta^2}\left(\min_{U\in\mathcal{S}(W)} I(Y;U|X)\right)^2 \tag{5.5}$$

and

$$\beta(\alpha) \geq \left(\min_{U:X\to Y\to U} I(Y;U|X)\right)^2 \int_{h=0}^{\infty} h \int_{\theta=0}^{1} f_X(\theta) e^{-hg(\theta)} d\theta dh \tag{5.6}$$

where $K, \delta > 0$ are constants and

$$g(\theta) \triangleq \left\{\frac{d}{d\Delta} - \left[\min_s \log\left(\int W_\alpha^s(y|\theta) W_\alpha^{1-s}(y|\theta+\Delta) dy\right)\right]\right\}_{\Delta=0} \tag{5.7}$$

is the first derivative of Chernoff information between the conditional densities W_α of the observation given $x = \theta$ and $x = \theta + \Delta$, evaluated at $\Delta = 0$. The minimums are taken over all non-trivial random variables to ensure that the conditional mutual information is non-zero.

Notice from the theorem, since $\beta(\alpha)$ is a finite constant, it implies that for a non-regular source distribution, in the limit of large sum rate, the distortion decays as $1/R^2$. This serves as an intermediate regime between the exponential decay of the discrete case [17] and $1/R$ decay of the quadratic Gaussian case [162]. This result can be summarized as the fact that sharing beliefs (uniform) is fundamentally easier (in terms of convergence rate) than sharing measurements (Gaussian), but sharing decisions is even easier (discrete). This shows the effect of the underlying source distribution on the asymptotic estimation performance. When the source has countably finite support set (discrete), an exponential decay is observed. On the other extreme, when the source has an unbounded support set (Gaussian), a $1/R$ decay is observed. Above result is for the source with bounded support (uniform, for example), and an intermediate result of $1/R^2$ decay is determined. This suggests the intuitive observation that as the number of possibilities for the source (support) increases, it gets more difficult to communicate the values.

One can also note the similarity in structure of the lower bound of $\beta(\alpha)$ in this problem with other CEO problems [17, 162]. Most notably, in all cases, there is a minimization of conditional mutual information. Also, the bound here depends on Chernoff information which serves as a divergence metric similar to the Kullback-Leibler divergence for the discrete case [17] and as an information metric similar to Fisher information for the quadratic Gaussian case [162].

5.2.1 Examples of Observation Channels Satisfying Property 5.2.1

Property 5.2.1 may seem a little opaque, so here we give an illustrative example of a family of observation channels that satisfy it.

Proposition 5.2.3. *A sufficient condition for an observation channel to satisfy Property 5.2.1 is when its density $W_\alpha(y_i|x)$ is given by a copula conditional density function*[2] *and has discontinuity*

---

[2] A copula is a multivariate probability distribution for which the marginal probability distribution of each variable

at end points.

Proof. Let the end points of observation channel $W_\alpha(y_i|x)$ be denoted by $e_l(x)$ and $e_u(x)$. Due to its discontinuity at the end points, we have the following

$$\lim_{y_i \to e_l(x) \text{ or } e_u(x)} W_\alpha(y_i|x) > 0. \tag{5.8}$$

Consider the test channel given by $U_i = Y_i + N$, where N is a k-peak noise for $k \geq 2$, $f_N(n) = \sum_{l=1}^{k} p_l \delta(n - n_l)$, where $\delta(\cdot)$ is the Dirac delta function, $\sum_{l=1}^{k} p_l = 1$, and $n_1 < n_2 < \cdots < n_k$. Then the conditional distribution of U given X, $f_{U|X}(u_i|x) = \sum_{l=1}^{k} p_l W_\alpha(u_i - n_l|x)$, has bounded support: $[e_l(x) + n_1, e_u(x) + n_k]$ and the values of $f_{U|X}(u_i|x)$ at the end points are given by

$$\lim_{u_i \to e_l(x)+n_1} f_{U|X}(u_i|x) \geq p_1 \lim_{y_i \to e_l(x)} W_\alpha(y_i|x) > 0 \tag{5.9}$$

and

$$\lim_{u_i \to e_u(x)+n_k} f_{U|X}(u_i|x) \geq p_k \lim_{y_i \to e_u(x)} W_\alpha(y_i|x) > 0. \tag{5.10}$$

This proves the proposition. □

We now provide a specific example from the above family of observation channels and explicitly show that it satisfies Property 5.2.1.

Example 5.2.4. *As a specific example, consider the case when the source* $X(t)$ *and the observations* $Y_i(t)$ *are marginally distributed with uniform distribution in* $(0, 1)$ *and Clayton copula model is used to model the channel between the source and the observations. The conditional distribution of* Y_i *given* X *is given by the following (for* $1/2 < \alpha < 1$*)*

$$W_\alpha(y_i|x) = \begin{cases} (1-\alpha)(xy_i)^{\alpha-1}\left(x^\alpha + y_i^\alpha - 1\right)^{1/\alpha - 2}, & \text{for } (1-x^\alpha)^{1/\alpha} \leq y_i \leq 1 \\ 0, & \text{otherwise.} \end{cases} \tag{5.11}$$

---

is uniform [99].

Using the test channel defined as $U_i = Y_i + N$*, where* N *is a* k*-peak noise for* $k \geq 2$*,* $f_N(n) = \sum_{l=1}^{k} p_l \delta(n - n_l)$*, where* $\delta(\cdot)$ *is the Dirac delta function and* $\sum_{l=1}^{k} p_l = 1$*, results in a density* $f_{U|X}(u_i|x)$ *given by (without loss of generality, assume* $n_1 < n_2 < \cdots < n_k$*):*

$$f_{U|X}(u_i|x) = \begin{cases} p_1(1-\alpha)(x(u_i - n_1))^{\alpha-1} \left(x^\alpha + (u_i - n_1)^\alpha - 1\right)^{1/\alpha - 2}, & \text{for } (1-x^\alpha)^{1/\alpha} + n_1 \leq u_i \leq 1 + n_1 \\ p_2(1-\alpha)(x(u_i - n_2))^{\alpha-1} \left(x^\alpha + (u_i - n_2)^\alpha - 1\right)^{1/\alpha - 2}, & \text{for } (1-x^\alpha)^{1/\alpha} + n_2 \leq u_i \leq 1 + n_2 \\ \vdots & \\ p_k(1-\alpha)(x(u_i - n_k))^{\alpha-1} \left(x^\alpha + (u_i - n_k)^\alpha - 1\right)^{1/\alpha - 2}, & \text{for } (1-x^\alpha)^{1/\alpha} + n_k \leq u_i \leq 1 + n_k \\ 0, & \text{otherwise.} \end{cases} \tag{5.12}$$

when $1 + n_l < (1 - x^\alpha)^{1/\alpha} + n_{l+1}$ *for* $l = 1, \ldots, k-1$ *to ensure the shifted versions of* $W_\alpha(y_i|x)$ *do not overlap.*[3]

We now show that $f_{U|X}(u_i|x)$ *given by* (5.12) *satisfies Property 5.2.1, which basically consists of two conditions on* $f_{U|X}(u|x)$*: bounded support and non-vanishing end points.* $f_{U|X}(u_i|x)$ *given in* (5.12) *has bounded support as* $(1 - x^\alpha)^{1/\alpha} + n_1 \leq u_i \leq 1 + n_k$ *(irrespective of whether the shifted versions overlap or not). Also, its values at the end points are given by:*

$$\lim_{u_i \to (1-x^\alpha)^{1/\alpha} + n_1} f_{U|X}(u_i|x) = p_1(1-\alpha)x^{\alpha-1}(1-x^\alpha)^{1-1/\alpha}(0)^{1/\alpha-2} \to \infty > 0 \tag{5.13}$$

as $\alpha > 1/2$ *and*

$$\lim_{u_i \to 1 + n_k} f_{U|X}(u_i|x) = p_k(1-\alpha)x^{-\alpha} > 0 \tag{5.14}$$

Hence, it satisfies Property 5.2.1.

Note the similarity between the form of this test channel and the random quantizer used by Fix for achieving the rate-distortion function of a uniform source [49].

As will be seen later in Sec. 5.3, the achievability involves use of a test channel, Slepian-Wolf encoding and decoding, and midrange estimation at the CEO. Fig. 5.1 provides a block diagram outlining these steps for the example considered here.

---

[3] It is straightforward to prove that the property holds even when there is overlap among the shifted versions.

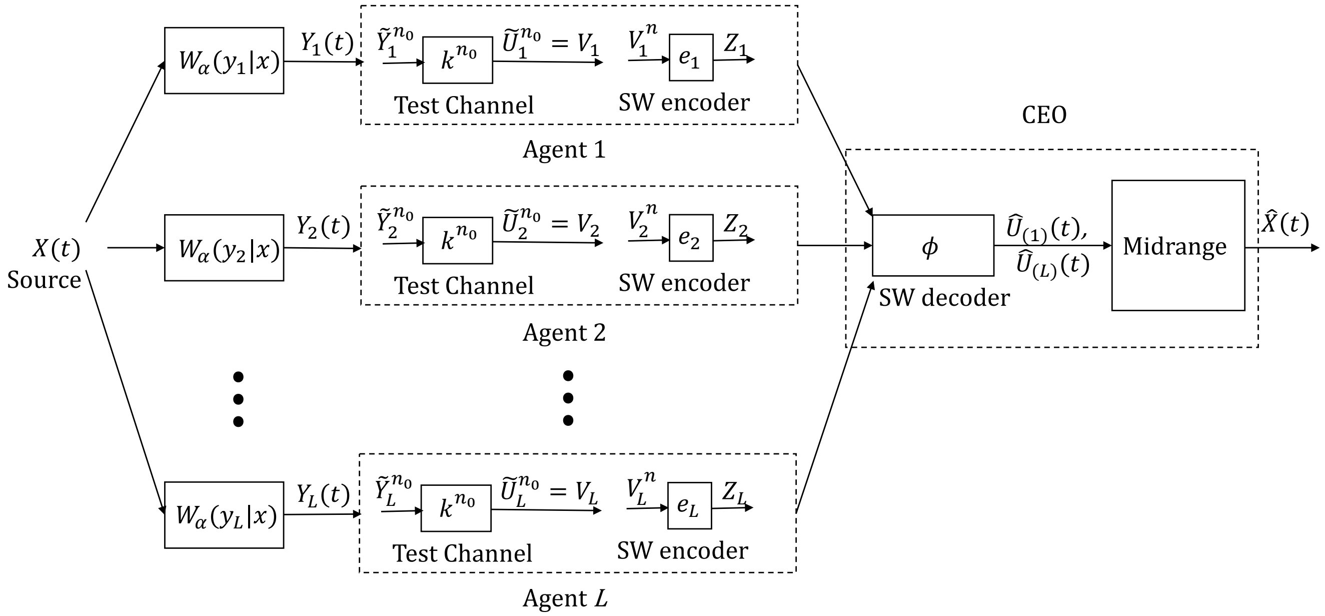


Fig. 5.1: A block diagram of the system model for Example 1 using the Clayton copula based observation channel $W_\alpha(y_i|x)$ given by (5.11) and test channel $U = Y + N$ where N is a k-peak noise. Here k^{n_0} represents the block code that approximates the test channel

5.3 Direct Coding Theorem

The structure of the achievable scheme is a layered architecture, with scalar quantization followed by Slepian-Wolf entropy coding, just like for the Gaussian CEO problem [162] and other source coding problems [124, 164, 172]. The following are key steps of the analysis: quantization of alphabets, codes that approximate the forward test channel, Slepian-Wolf encoding and decoding, and estimation at the CEO.

Every agent uses a two-stage encoding scheme. In the first stage, a block of observations are mapped to codewords from a codebook which is identical for all agents. The second stage is an index encoder that performs Slepian-Wolf encoding of the codewords [39, 129]. For decoding, the CEO first performs index decoding to determine the L codewords corresponding to each of the agents, and then estimates the source value at each instant based on a midrange estimator [10, 100].

The key aspect of the proof is the choice of forward test channel which is characterized by an auxiliary random variable U. Choose the test channel from Y to U, denoted by $Q(u|y)$ where $U \in \mathcal{S}(W)$, so as to induce a distribution $f_{U|X}(u|x)$ which satisfies Property 5.2.1.

5.3.1 Quantization of Alphabets

To design the coding scheme, start by quantizing the continuous alphabets. Denote by $\tilde{X}$, $\tilde{Y}$, and $\tilde{U}$ the quantized versions of random variables X, Y, and U, respectively. Their corresponding alphabets are denoted by $\tilde{\mathcal{X}}$, $\tilde{\mathcal{Y}}$, and $\tilde{\mathcal{U}}$ respectively. Conditions to be satisfied by the quantization are as follows:

$$E(U-\tilde{U})^2 \leq \delta_0 \tag{5.15}$$

$$|I(Y;U) - I(\tilde{Y},\tilde{U})| \leq \delta_1 \tag{5.16}$$

$$|I(X;U) - I(\tilde{X},\tilde{U})| \leq \delta_2, \tag{5.17}$$

where $\delta_j > 0$, for $j = 0, 1, 2$. There exist quantization schemes that achieve each of these above constraints individually: (5.15) from that fact that $EU^2 < \infty$, and (5.16) and (5.17) from the definition of mutual information for arbitrary ensembles [44]. Therefore, a common refinement of the quantization schemes that achieve (5.15)–(5.17) separately will satisfy them simultaneously. This quantization induces a corresponding joint probability distribution for the quantized versions $\tilde{X}$, $\tilde{Y}$, and $\tilde{U}$:

$$\begin{aligned}
P_{\tilde{Y},\tilde{U}}(\tilde{y},\tilde{u}) &= \int_{\{(y,u):\text{quant}(y)=\tilde{y},\text{quant}(u)=\tilde{u}\}} f_{Y,U}(y,u)dydu \\
P_{\tilde{X},\tilde{U}}(\tilde{x},\tilde{u}) &= \int_{\{(x,u):\text{quant}(x)=\tilde{x},\text{quant}(u)=\tilde{u}\}} f_X(x)f_{U|X}(u|x)dxdu \\
\tilde{W}_\alpha(\tilde{y}|x) &= \int_{\{y:\text{quant}(y)=\tilde{y}\}} W_\alpha(y|x)dy \\
Q(\tilde{u}|\tilde{y}) &= \frac{P_{\tilde{Y},\tilde{U}}(\tilde{y},\tilde{u})}{P_{\tilde{Y}}(\tilde{y})}.
\end{aligned}$$

Note that any letters $\tilde{y}$ of zero measure are removed from $\tilde{\mathcal{Y}}$.

5.3.2 Codes That Approximate the Test Channel

The encoding scheme works on the quantized version $\tilde{Y}_i$. The basic idea is to build a block code between the quantized versions $\tilde{Y}_i$ and $\tilde{U}_i$, and show that the designed block code approximates the test channel $Q(u|y)$ that arises from satisfying Property 5.2.1. Let k^{n_0} be a block code of length n_0 from $\tilde{\mathcal{Y}}^{n_0}$ to $\tilde{\mathcal{U}}^{n_0}$. This map, k^{n_0}, induces the following joint distribution between the blocks $\tilde{Y}^{n_0} = [\tilde{Y}(1), \ldots, \tilde{Y}(n_0)]$ and $\tilde{U}^{n_0} = [\tilde{U}(1), \ldots, \tilde{U}(n_0)]$:

$$\hat{P}^{n_0}(\tilde{Y}^{n_0} = \tilde{y}^{n_0}, \tilde{U}^{n_0} = \tilde{u}^{n_0}) = P_{\tilde{Y}^{n_0}}(\tilde{y}^{n_0}) \mathbb{1}_{\{k^{n_0}(\tilde{y}^{n_0}) = \tilde{u}^{n_0}\}},$$

where $\mathbb{1}_{\mathcal{A}}$ is the indicator function which is 1 when the event $\mathcal{A}$ is true and 0 otherwise. Also, the corresponding marginals and conditionals are given by

$$\hat{P}(\tilde{Y}(t) = \tilde{y}, \tilde{U}(t) = \tilde{u}) = E_{P_{\tilde{Y}^n}} \mathbb{1}_{\left\{\tilde{U}(t) = \tilde{u}, \tilde{Y}(t) = \tilde{y}\right\}}$$

$$\hat{Q}(\tilde{U}(t) = \tilde{u} | \tilde{Y}(t) = \tilde{y}) = \frac{\hat{P}(\tilde{Y}(t) = \tilde{y}, \tilde{U}(t) = \tilde{u})}{P_{\tilde{Y}}(\tilde{Y}(t) = \tilde{y})}.$$

Now the existence of a block code $k^{n_0} : \tilde{\mathcal{Y}}^{n_0} \to \tilde{\mathcal{U}}^{n_0}$ which approximates a test channel $Q(u|y)$ arising from Property 5.2.1 follows from [162, Proposition 3.1], which is stated here without proof.

Proposition 5.3.1 ([162]). *For every* $\epsilon_0, \delta_3 > 0$*, there exists a deterministic map* $k^{n_0} : \tilde{\mathcal{Y}}^{n_0} \to \tilde{\mathcal{U}}^{n_0}$ *with the range cardinality* M *such that*

$$\frac{1}{n_0} \log M \leq I(Y; U) + \delta_3 \tag{5.18}$$

and

$$\sum_{\tilde{u} \in \tilde{\mathcal{U}}} |\hat{Q}(\tilde{U}(t) = \tilde{u} | x) - Q(\tilde{U}(t) = \tilde{u} | x)| \leq \frac{\epsilon_0}{|\tilde{\mathcal{X}}|}$$

for all $t = 1, \ldots, n_0$ *and all real* x.

5.3.3 Encoding and Decoding

The encoding is performed in two stages: in the first stage, the agents use the identical deterministic mapping k^{n_0} of Proposition 5.3.1 to encode their quantized observation block $\tilde{Y}_i^{n_0}$ into codewords $\tilde{U}_i^{n_0}$; and in the second stage, Slepian-Wolf encoding [129] is used to encode the index of each agent's codeword $\tilde{U}_i^{n_0}$. Let the index of codeword $\tilde{U}_i^{n_0}$ in the codebook be denoted by V_i, for $i = 1, \ldots, L$. The index is used to represent the codeword due to the one-to-one correspondence between the index and the codeword, therefore, we have $V_i = \tilde{U}_i^{n_0}$. Note that $V_1, \ldots, V_L$ are correlated and Slepian-Wolf encoding of the indices is used to remove that correlation across agents. This is done by index encoding the n-length block of indices of agent i, represented as $V_i^n = [V_i(1), \ldots, V_i(n)]$, where $V_i(t)$ is the tth component of the n-block of the indices of agent i. This block of indices is then mapped to a smaller index set using a mapping $e_i : \tilde{\mathcal{U}}^{nn_0} \to \{0, \ldots, N_i - 1\}$, for $i = 1, \ldots, L$, where N_i and n are chosen to be sufficiently large to ensure a negligible decoding error. The sum rate per source symbol is given by

$$R = \frac{1}{nn_0} \sum_{i=1}^{L} \log N_i.$$

Therefore, the complete encoder is given by $h_i = e_i \circ k^{n_0} : \tilde{\mathcal{Y}}^{nn_0} \to \{0, 1, 2, \ldots, N_i - 1\}$, where '$\circ$' is the composition operator. Let the output of this encoder be represented by $Z_i = h_i(\tilde{Y}_i^{nn_0}) \in \{0, 1, \ldots, N_i - 1\}$.

The CEO receives the indices $Z_1, \ldots, Z_L$ corresponding to the L agents. It first recovers the block of indices $\hat{V}_i^n$, for all i using a mapping $\phi : \prod_{i=1}^{L} \{0, 1, \ldots, N_i - 1\} \to \prod_{i=1}^{L} \tilde{\mathcal{U}}_i^{nn_0}$. The output of this decoder, represented as $\hat{V}_i^n = [\hat{V}_i(1), \ldots, \hat{V}_i(n)] = [\hat{U}_i^{n_0}(1), \ldots, \hat{U}_i^{n_0}(n)]$, is the decoded super codeword and $\hat{U}_i^{n_0}(t)$ is the decoded version of $\tilde{U}_i^{n_0}(t)$. From the Slepian-Wolf theorem (cf. [162, Proposition 3.2]), there exist encoders $\{e_i\}$ and a decoder ϕ such that the codewords can be recovered with negligible error probability for sufficiently large block size n.

Proposition 5.3.2 ([162]). *For every $\epsilon_1, \lambda > 0$, there exist sufficiently large L, n, and index en-*

coders $e_1, \ldots, e_L$ *and index decoder* ϕ *such that*

$$\frac{R}{L} \leq \frac{1}{n_0} H(\tilde{U}^{n_0} | \tilde{X}^{n_0}) + \epsilon_1, \tag{5.19}$$

$$\Pr\{(\hat{U}_1^{n_0}, \ldots, \hat{U}_L^{n_0}) \neq (\tilde{U}_1^{n_0}, \ldots, \tilde{U}_L^{n_0}) \leq \lambda\}, \tag{5.20}$$

where $\tilde{X}^{n_0} = [\tilde{X}(1), \ldots, \tilde{X}(n_0)]$.

5.3.4 Further Analysis of Code Rate

Note that the bound on sum rate per agent R/L in (5.19) is in terms of the distributions of $\tilde{U}$ and $\tilde{X}$. By further analyzing the code rate, a bound can be determined which is a function of the distributions of the unquantized versions, X and U. For this, the closeness of the marginal distribution induced by the encoding function k^{n_0} to the test channel statistics is used to bound the entropy terms. Let $H(\tilde{X})$ denote the entropy of the quantized random variable $\tilde{X}$, then

$$\begin{aligned}
\frac{1}{n_0} H(\tilde{U}^{n_0} | \tilde{X}^{n_0}) &= \frac{1}{n_0} (H(\tilde{U}^{n_0}, \tilde{X}^{n_0}) - H(\tilde{X}^{n_0})) && (5.21)\\
&= \frac{1}{n_0} H(\tilde{U}^{n_0}) + \frac{1}{n_0} H(\tilde{X}^{n_0} | \tilde{U}^{n_0}) - \frac{1}{n_0} H(\tilde{X}^{n_0}) && (5.22)\\
&= \frac{1}{n_0} H(\tilde{U}^{n_0}) + \frac{1}{n_0} H(\tilde{X}^{n_0} | \tilde{U}^{n_0}) - \frac{1}{n_0} \sum_{t=1}^{n_0} H(\tilde{X}(t)) && (5.23)\\
&\leq \frac{1}{n_0} \log M + \frac{1}{n_0} H(\tilde{X}^{n_0} | \tilde{U}^{n_0}) - \frac{1}{n_0} \sum_{t=1}^{n_0} H(\tilde{X}(t)) && (5.24)\\
&\leq I(Y;U) + \frac{1}{n_0} H(\tilde{X}^{n_0} | \tilde{U}^{n_0}) - \frac{1}{n_0} \sum_{t=1}^{n_0} H(\tilde{X}(t)) + \delta_3 && (5.25)
\end{aligned}$$

where (5.23) is due to the independent nature of source $X(t)$ over time, (5.24) is by upper bounding $H(\tilde{U}^{n_0})$ by the logarithm of number of codewords M, and (5.25) follows from (5.18). Next,

$H(\tilde{X}^{n_0}|\tilde{U}^{n_0})$ can be further bounded as:

$$\begin{aligned} H(\tilde{X}^{n_0}|\tilde{U}^{n_0}) &= \sum_{t=1}^{n_0} H(\tilde{X}(t)|\tilde{U}^{n_0}, \tilde{X}(1), \ldots, \tilde{X}(t-1)) \\ &\leq \sum_{t=1}^{n_0} H(\tilde{X}(t)|\tilde{U}(t)) \qquad (5.26) \end{aligned}$$

using the fact that conditioning only reduces entropy. Therefore, we have

$$\begin{aligned} \frac{1}{n_0} H(\tilde{U}^{n_0}|\tilde{X}^{n_0}) &\leq I(Y;U) + \frac{1}{n_0} H(\tilde{X}^{n_0}|\tilde{U}^{n_0}) - \frac{1}{n_0}\sum_{t=1}^{n_0} H(\tilde{X}(t)) + \delta_3 \qquad (5.27) \\ &\leq I(Y;U) + \frac{1}{n_0}\sum_{t=1}^{n_0} H(\tilde{X}(t)|\tilde{U}(t)) - \frac{1}{n_0}\sum_{t=1}^{n_0} H(\tilde{X}(t)) + \delta_3 \\ &= I(Y;U) - \frac{1}{n_0}\sum_{t=1}^{n_0} I(\tilde{X}(t), \tilde{U}(t)) + \delta_3 \qquad (5.28) \\ &\leq I(Y;U) - I(\tilde{X}, \tilde{U}) + \delta_3 + 2\epsilon_0 \log \frac{|\tilde{\mathcal{U}}||\tilde{\mathcal{X}}|}{\epsilon_0} \qquad (5.29) \end{aligned}$$

where (5.29) is due to the t-symmetry of the encoder and [162, Proposition A.3].

Thus, using (5.16):

$$\begin{aligned} \frac{1}{n_0} H(\tilde{U}^{n_0}|\tilde{X}^{n_0}) &\leq I(Y;U) - I(\tilde{X}, \tilde{U}) + \delta_3 + 2\epsilon_0 \log \frac{|\tilde{\mathcal{U}}||\tilde{\mathcal{X}}|}{\epsilon_0} \qquad (5.30) \\ &\leq I(Y;U) - I(X;U) + \delta_2 + \delta_3 + 2\epsilon_0 \log \frac{|\tilde{\mathcal{U}}||\tilde{\mathcal{X}}|}{\epsilon_0}. \qquad (5.31) \end{aligned}$$

Due to the Markov chain relationship $X \rightarrow Y \rightarrow U$, the right hand side can be further simplified to

$$\frac{1}{n_0} H(\tilde{U}^{n_0}|\tilde{X}^{n_0}) \leq I(Y;U|X) + \delta_2 + \delta_3 + 2\epsilon_0 \log \frac{|\tilde{\mathcal{U}}||\tilde{\mathcal{X}}|}{\epsilon_0} \qquad (5.32)$$

By choosing $\delta_2, \delta_3, \epsilon_0, \epsilon_2$ such that

$$\delta_2 + \delta_3 + 2\epsilon_0 \log \frac{|\tilde{\mathcal{U}}||\tilde{\mathcal{X}}|}{\epsilon_0} < \epsilon_2,$$

we have

$$\frac{R}{L} \leq I(Y;U|X) + \epsilon_1 + \epsilon_2. \tag{5.33}$$

Having determined a bound on sum rate, the next step is to bound the minimum quadratic distortion.

5.3.5 Estimation Scheme

The CEO, after decoding the codewords sent by the agents $(\hat{U}_1^{n_0}, \dots, \hat{U}_L^{n_0})$, estimates the source $X(t)$ on an instant-by-instant basis. Since the range of $U_i(t)$ depends on $X(t)$, we first estimate the midrange of data $\hat{U}_1(t), \dots, \hat{U}_L(t)$. The midrange estimator [10, 41, 100, 120] is the maximally efficient estimator for the center of a uniform distribution. The midrange estimator also seems to work well for estimating the location parameter of other distributions of bounded support and it is more effective than the sample mean for many distributions such as the cosine distribution, parabolic distribution, rectangular distribution, and inverted parabolic distribution [120], though the best estimator depends on the distribution of the source that is to be estimated. For these reasons, the midrange estimator is used here.

After estimating the midrange of data, using the inverse function $l(\cdot)$ as follows (cf. Property 5.2.1),

$$\hat{X}(t) = 2l\left(\frac{\hat{U}_{(1)}(t) + \hat{U}_{(L)}(t)}{2}\right), \tag{5.34}$$

one gets an estimate of $X(t)$. Here $\hat{U}_{(i)}(t)$ are the order statistics of $\hat{U}_i(t)$ [43]. Note that $E\left[\frac{\hat{U}_{(1)}+\hat{U}_{(L)}}{2}\right] = E\left[\frac{a(X)+b(X)}{2}\right]$.

An upper bound on the distortion can now be derived, following a method similar to Açkay, et al. [1]:

$$\begin{aligned}
& E[\hat{X}(t)-X(t)]^2 \\
\leq\ & K^2 E\left[\left(\frac{\hat{U}_{(1)}(t)+\hat{U}_{(L)}(t)}{2}-\frac{a(X(t))+b(X(t))}{2}\right)^2\right] \\
=\ & K^2 E_X E_{U|X}\left[\left(\frac{\hat{U}_{(1)}(t)+\hat{U}_{(L)}(t)}{2}-\frac{a(X(t))+b(X(t))}{2}\right)^2 \middle| X(t)\right] \qquad (5.35) \\
\leq\ & 2K^2 E_X E_{U|X}\left[\left(\frac{U_{(1)}(t)+U_{(L)}(t)}{2}-\frac{a(X(t))+b(X(t))}{2}\right)^2 \middle| X(t)\right]+\epsilon_3 \qquad (5.36)
\end{aligned}$$

where $U_{(i)}(t)$ are the order statistics of $U_i(t)$; the first inequality is due to Lipschitz continuity of the function $l(\cdot)$ with Lipschitz constant K, and (5.36) follows from Proposition A.6.1 in the Appendix.

Now we evaluate the main term in (5.36); for notational simplicity, we drop the dependence on t and the dependence of $a(\cdot)$ and $b(\cdot)$ on X. However, we need to be aware of the dependence of the limits a and b on the unknown X. As $f_{U|X}(u|x)$ does not vanish at the endpoints, there exist ϵ and δ such that $f_{U|X}(u|x) \geq \delta$ for $a \leq u \leq a+\epsilon$ and $b-\epsilon \leq u \leq b$. Now,

$$\begin{aligned}
& E_{U|X}\left[\left(\frac{U_{(1)}+U_{(L)}}{2}-\frac{a+b}{2}\right)^2 \middle| X\right] \\
& = \int_{u_{(1)}>a+\epsilon \text{ or } u_{(L)}<b-\epsilon}\left(\frac{u_{(1)}+u_{(L)}}{2}-\frac{a+b}{2}\right)^2 f_{u_{(1)},u_{(L)}|X} du_{(1)} du_{(L)} \\
& + \int_{u_{(1)}<a+\epsilon \text{ and } u_{(L)}>b-\epsilon}\left(\frac{u_{(1)}+u_{(L)}}{2}-\frac{a+b}{2}\right)^2 f_{u_{(1)},u_{(L)}|X} du_{(1)} du_{(L)}. \qquad (5.37)
\end{aligned}$$

Since $(u_{(1)}+u_{(L)})/2 \in [a,b]$:

$$\left(\frac{u_{(1)}+u_{(L)}}{2}-\frac{a+b}{2}\right)^2 \leq \left(\frac{b-a}{2}\right)^2$$

and the first term on right side of (5.37) can be bounded as:

$$\int_{u_{(1)}>a+\epsilon \text{ or } u_{(L)}<b-\epsilon} \left(\frac{u_{(1)}+u_{(L)}}{2} - \frac{a+b}{2}\right)^2 f_{u_{(1)},u_{(L)}|X} du_{(1)} du_{(L)}$$
$$\leq 2^{-2}(b-a)^2 \Pr\left\{u_{(1)} > a+\epsilon \text{ or } u_{(L)} < b-\epsilon|X\right\}.$$

Since the $\{U_i\}$ are conditionally independent given X, further simplification can be done as:

$$\Pr\left\{u_{(1)} > a+\epsilon \text{ or } u_{(L)} < b-\epsilon|X\right\}$$
$$\leq \Pr\left\{u_{(1)} > a+\epsilon|X\right\} + \Pr\left\{u_{(L)} < b-\epsilon|X\right\} \tag{5.38}$$
$$= \prod_{i=1}^{L} \Pr\{u_i > a+\epsilon|X\} + \prod_{i=1}^{L} \Pr\{u_i < b-\epsilon|X\} \tag{5.39}$$
$$= \prod_{i=1}^{L}(1-\Pr\{u_i \leq a+\epsilon|X\}) + \prod_{i=1}^{L}(1-\Pr\{u_i \geq b-\epsilon|X\}). \tag{5.40}$$

Since, $f_{U|X}(u|x) \geq \delta$ for $a \leq u \leq a+\epsilon$ and $b-\epsilon \leq u \leq b$, $\Pr\{u_i \leq a+\epsilon|X\} \geq \delta\epsilon$ and $\Pr\{u_i \geq b-\epsilon|X\} \geq \delta\epsilon$. Therefore,

$$\Pr\left\{u_{(1)} > a+\epsilon \text{ or } u_{(L)} < b-\epsilon|X\right\}$$
$$\leq \prod_{i=1}^{L}(1-\Pr\{u_i \leq a+\epsilon|X\}) + \prod_{i=1}^{L}(1-\Pr\{u_i \geq b-\epsilon|X\})$$
$$\leq \prod_{i=1}^{L}(1-\delta\epsilon) + \prod_{i=1}^{L}(1-\delta\epsilon) \tag{5.41}$$
$$\leq 2(1-\delta\epsilon)^L. \tag{5.42}$$

To evaluate the second term in the right side of (5.37), define the following variables:

$$\xi = L(1-F_{U|X}(u_{(L)})),\ b-\epsilon \leq u_{(L)} \leq b \tag{5.43}$$

$$\eta = LF_{U|X}(u_{(1)}),\ a \leq u_{(1)} \leq a+\epsilon, \tag{5.44}$$

where $F_{U|X}$ is the conditional cumulative distribution function of U given X. These variables have the following marginal and joint densities [1]:

$$f_\xi(s) = f_\eta(s) = \left(1 - \frac{s}{L}\right)^{L-1}, \quad 0 \le s \le L \tag{5.45}$$

$$f_{\xi,\eta}(s_1, s_2) = \frac{L-1}{L}\left(1 - \frac{s_1 + s_2}{L}\right)^{L-2}, \quad s_1, s_2 \ge 0, \quad s_1 + s_2 \le L.$$

Also, as $L \to \infty$, ξ and η become independent and $f_\xi(s), f_\eta(s) \to e^{-s}$.

From the above definitions,

$$\xi = L \int_{u_{(L)}}^{b} f_{u|x} du \ge \delta L(b - u_{(L)}), \tag{5.46}$$

$$\eta = L \int_{a}^{u_{(1)}} f_{u|x} du \ge \delta L(u_{(1)} - a), \tag{5.47}$$

provided $u_{(1)} \le a + \epsilon$ and $b - \epsilon \le u_{(L)}$. Therefore, for the second term, we have

$$\left|\frac{u_{(1)} + u_{(L)}}{2} - \frac{a+b}{2}\right|^2 = \frac{1}{4}\left|(u_{(1)} - a) - (b - u_{(L)})\right|^2 \tag{5.48}$$

$$\le \frac{1}{4}\left[\left|u_{(1)} - a\right|^2 + \left|b - u_{(L)}\right|^2\right] \tag{5.49}$$

$$\le \frac{\xi^2 + \eta^2}{4\delta^2 L^2}, \tag{5.50}$$

where the fact that $|A - B|^2 \le A^2 + B^2$ for $A, B > 0$, is used.

Now using the inequalities that have been developed, one can bound the distortion in (5.36) as:

$$D(L, R) \le 2K^2 E_X\left[\tfrac{(b-a)^2(1-\delta\epsilon)^L}{2} + \tfrac{1}{4\delta^2 L^2}\int_0^{L(1-F_{U|X}(b-\epsilon))}\int_0^{LF_{U|X}(a+\epsilon)} (s_1^2 + s_2^2) f_{\xi,\eta}(s_1, s_2) ds_1 ds_2\right] + \epsilon_3. \tag{5.51}$$

Using (5.33) and (5.51), we get:

$$R^2D(L,R) \le L^2I^2(Y;U|X)\Bigg(2K^2E_X\Bigg[\frac{(b-a)^2(1-\delta\epsilon)^L}{2} + \frac{1}{4\delta^2L^2}\int_0^{L(1-F_{U|X}(b-\epsilon))}\int_0^{LF_{U|X}(a+\epsilon)}(s_1^2+s_2^2)f_{\xi,\eta}(s_1,s_2)ds_1ds_2\Bigg]+\epsilon_3\Bigg).$$

By taking limits $L, R \to \infty$, we have:

$$\begin{aligned}
\beta(\alpha) &= \lim_{L,R\to\infty} R^2D(L,R) \\
&\le I^2(Y;U|X)\left(2K^2E_X\left[\frac{1}{4\delta^2}\int_0^\infty\int_0^\infty(s_1^2+s_2^2)\lim_{L\to\infty}f_{\xi,\eta}(s_1,s_2)ds_1ds_2\right]\right) && (5.52)\\
&= I^2(Y;U|X)\left(2K^2E_X\left[\frac{1}{4\delta^2}\int_0^\infty\int_0^\infty(s_1^2+s_2^2)e^{-s_1}e^{-s_2}ds_1ds_2\right]\right)\\
&= I^2(Y;U|X)\left(2K^2E_X\left[\frac{1}{2\delta^2}\int_0^\infty s^2e^{-s}ds\right]\right) && (5.53)\\
&= \frac{2K^2}{\delta^2}I^2(Y;U|X) > 0 && (5.54)
\end{aligned}$$

where U is chosen to satisfy Property 5.2.1 and $K > 0$ is a constant. Therefore,

$$\beta(\alpha) \le \frac{2K^2}{\delta^2}\left(\min_{U\in\mathcal{S}(W)} I(Y;U|X)\right)^2. \tag{5.55}$$

This concludes the achievability proof. Note that the bound only depends on the conditional mutual information $I(Y;U|X)$ which corresponds to the compression of the observation noise. The compression of the source X does not appear in the bound, since such a term vanishes because the number of agents L grows without bound.

5.4 Converse Coding Theorem

The converse for the quadratic non-regular CEO problem is similar in structure to the converse for the quadratic Gaussian CEO [162] and the discrete CEO problem [17]. The proof uses a lower bound on the distortion function similar to the Bayesian Cramér-Rao lower bound used in [162]. However, note that the source distribution herein does not satisfy the regularity conditions required for using the Cramér-Rao bound [145]. Therefore, a version of the extended Chazan-Zakai-Ziv bound [14, 15, 29] is used, which is first stated here without proof.

Lemma 5.4.1. *For estimating a random scalar parameter* $x \sim f_X(x)$ *with support on* $[0, T]$ *using data* $\mathbf{z} = [z_1, \ldots, z_k]$ *with conditional distribution* $f(\mathbf{z}|x)$*, the MSE between* x *and* $\hat{x}(\mathbf{z})$ *is bounded as follows:*

$$E(x - \hat{x}(\mathbf{z}))^2 \geq \frac{1}{2T} \int_{h=0}^{T} h \left[\int_{\theta=0}^{T-h} (f_X(\theta) + f_X(\theta + h)) P_{min}(\theta, \theta + h) d\theta \right] dh, \tag{5.56}$$

where $P_{min}(\theta, \theta + h)$ *is the minimum error probability corresponding to the following binary hypothesis testing problem:*

$$\begin{aligned} H_0 &: \mathbf{z} \sim f(\mathbf{z}|x), \quad x = \theta, \qquad \Pr(H_0) = \frac{f_X(\theta)}{f_X(\theta) + f_X(\theta + h)}, \\ H_1 &: \mathbf{z} \sim f(\mathbf{z}|x), \quad x = \theta + h, \qquad \Pr(H_1) = \frac{f_X(\theta + h)}{f_X(\theta) + f_X(\theta + h)}. \end{aligned}$$

The above Chazan-Zakai-Ziv bound falls under the family of Ziv-Zakai bounds. Ziv-Zakai bounds have been shown to be useful bounds for all regions of operation unlike other bounds (for example, Cramér-Rao bound) that have limited applicability [14]. This family of bounds build on the original Ziv-Zakai bound [174] and have the advantage of being independent of bias and very tight in most cases. A detailed study of this family of bounds can be found in [14].

Note that this lemma bounds the performance of an estimation problem in terms of the performance of a sequence of detection problems. Therefore, as shall be seen later, one gets Chernoff information rather than Fisher information as seen in the estimation problem in the quadratic Gaussian CEO [162].

Using Lemma 5.4.1, the converse is now proved. Let $\{\mathcal{C}_i^n\}_{i=1}^L$ be L codes of block length n, corresponding to the L agents, with respective rates $R_1, R_2, \ldots, R_L$. The genie-aided approach is used to determine the lower bound as follows: Let the CEO implement n estimators O_t for $t = 1, \ldots, n$ where O_t estimates $X(t)$ given all components of the source word x^n except $x(t)$. Recall the definition of $X^n = [X(1), \ldots, X(n)]$ and further define $Y_i^n = [Y_i(1), \ldots, Y_i(n)]$. This gives

$$\begin{aligned}
nR_i &= \log |\mathcal{C}_i^n| \\
&\geq I(Y_i^n; C_i | X^n) \\
&= \sum_{t=1}^{n} I(Y_i(t); C_i | Y_i^{t-1}, X^n) && (5.57)\\
&= \sum_{t=1}^{n} \left[h\left(Y_i(t) | Y_i^{t-1}, X^n\right) - h\left(Y_i(t) | C_i, Y_i^{t-1}, X^n\right)\right] \\
&= \sum_{t=1}^{n} \left[h\left(Y_i(t) | X^n\right) - h\left(Y_i(t) | C_i, Y_i^{t-1}, X^n\right)\right] && (5.58)\\
&\geq \sum_{t=1}^{n} [h(Y_i(t)|X^n) - h(Y_i(t)|C_i, X^n)] && (5.59)\\
&= \sum_{t=1}^{n} I(Y_i(t); C_i | X^n),
\end{aligned}$$

where X is the generic source random variable, Y_i is the noisy version of X as observed by agent i; (5.57) is from the product rule of mutual information, (5.58) is due to the independence of $Y(t)$ across time, and (7.6) follows since conditioning only reduces entropy.

Hence, a lower bound on the sum rate R is given as follows:

$$R \geq \frac{1}{n} \sum_{t=1}^{n} \sum_{i=1}^{L} I(Y_i(t); C_i | X^n).$$

Define $\breve{X}_t = (X_1, \ldots, X_{t-1}, X_{t+1}, \ldots, X_n)$ and let $U_i(t, \breve{x}_t)$ be a random variable whose joint distribution with $X(t)$ and $Y_i(t)$ is:

$$\begin{aligned}
&\Pr\{x \le X(t) \le x+dx, y \le Y_i(t) \le y+dy, U_i(t,\breve{x}_t)=c\} \\
&\quad = f_X(x)W_\alpha(y|x)\Pr(C_i=c|Y_i(t)=y, X(t)=x, \breve{X}_t=\breve{x}_t)dxdy \\
&\quad = f_X(x)W_\alpha(y|x)\Pr(C_i=c|Y_i(t)=y, \breve{X}_t=\breve{x}_t)dxdy,
\end{aligned}$$

since the codeword C_i depends on $X(t)$ only through $Y_i(t)$. Therefore, for each i and any fixed $\breve{x}_t$, we have the Markov chain relationship $X(t) \to Y_i(t) \to U_i(t,\breve{x}_t)$. Now, we can express the lower bound on R as

$$R \ge \frac{1}{n}\sum_{t=1}^{n}\sum_{i=1}^{L} E_{\breve{X}_t} I(Y_i(t); U_i(t,\breve{X}_t)|X(t)). \tag{5.60}$$

Note that in order to find a lower bound on $\beta(\alpha)$, we consider the best case where the CEO knows $C_1,\ldots,C_L$ and $\breve{x}_t$, i.e., the CEO uses an estimator $\hat{X}(C_1,\ldots,C_L,\breve{x}_t)$. Using the Chazan-Zakai-Ziv bound (Lemma 5.4.1):

$$E(X(t)-\hat{X}_t)^2 \ge \frac{1}{2}\int_{h=0}^{1} h\left[\int_{\theta=0}^{1-h}(f_X(\theta)+f_X(\theta+h))P_{min,t}(\theta,\theta+h)d\theta\right]dh \tag{5.61}$$

where $P_{min,t}(\theta,\theta+h)$ is the minimum achievable error probability, using data $Y_1(t),\ldots,Y_L(t)$ from the L agents, to differentiate between $X(t)=\theta$ and $X(t)=\theta+h$.

Therefore, from the definition of $D(L,R)$:

$$\begin{aligned}
D(L,R) &= \frac{1}{n}\sum_{t=1}^{n} E(X(t)-\hat{X}_t)^2 \\
&\ge \frac{1}{2n}\sum_{t=1}^{n}\left[\int_{h=0}^{1} h\left[\int_{\theta=0}^{1-h}(f_X(\theta)+f_X(\theta+h))P_{min,t}(\theta,\theta+h)d\theta\right]dh\right] \\
&\ge \frac{1}{2nL^2}\sum_{t=1}^{n}\left[\int_{h=0}^{1} hL\left[\int_{\theta=0}^{1-h}(f_X(\theta)+f_X(\theta+h))P_{min,t}(\theta,\theta+h)d\theta\right]d(hL)\right],
\end{aligned}$$

where we have multiplied and divided the right side by L^2. Now, using a change of variables

$\tilde{h} = hL$:

$$
\begin{aligned}
&D(L,R)\\
&\geq \frac{1}{2nL^2}\sum_{t=1}^{n}\int_{(hL)=0}^{L}(hL)\int_{\theta=0}^{1-(hL)/L}(f_X(\theta)+f_X(\theta+(hL)/L))P_{min,t}(\theta,\theta+(hL)/L)d\theta d(hL)\\
&= \frac{1}{2nL^2}\sum_{t=1}^{n}\left[\int_{\tilde{h}=0}^{L}\tilde{h}\left[\int_{\theta=0}^{1-\tilde{h}/L}(f_X(\theta)+f_X(\theta+\tilde{h}/L))P_{min,t}(\theta,\theta+\tilde{h}/L)d\theta\right]d\tilde{h}\right]\\
&\geq \frac{1}{2L^2}\frac{1}{\frac{1}{n}\sum_{t=1}^{n}\frac{1}{\left[\int_{\tilde{h}=0}^{L}\tilde{h}\left[\int_{\theta=0}^{1-\tilde{h}/L}(f_X(\theta)+f_X(\theta+\tilde{h}/L))P_{min,t}(\theta,\theta+\tilde{h}/L)d\theta\right]d\tilde{h}\right]}},
\end{aligned}
\tag{5.62}
$$

where the last step is due to the inequality of arithmetic and harmonic means.

Note that, although not explicit, $D(L,R)$ does depend on R. This dependence is implicitly visible via n (see (5.60)). Therefore, as can be observed below in (5.63), the product of R^2 and $D(L,R)$ results in a positive constant that is independent of R and does not vanish as $L \to \infty$.

Using (5.60) and (5.62), one gets the following expression:

$$
\begin{aligned}
&R^2 D(L,R)\\
&\geq \frac{1}{n^2}\frac{n}{2L^2}\frac{\left(\sum_{t=1}^{n}\sum_{i=1}^{L}E_{\breve{X}_t}I(Y_i(t);U_i(t,\breve{X}_t)|X(t))\right)^2}{\sum_{t=1}^{n}\left[\int_{\tilde{h}=0}^{L}\tilde{h}\left[\int_{\theta=0}^{1-\tilde{h}/L}(f_X(\theta)+f_X(\theta+\tilde{h}/L))P_{min,t}(\theta,\theta+\tilde{h}/L)d\theta\right]d\tilde{h}\right]^{-1}}\\
&= \frac{1}{2nL^2}\frac{\left(\sum_{t=1}^{n}\sum_{i=1}^{L}E_{\breve{X}_t}I(Y_i(t);U_i(t,\breve{X}_t)|X(t))\right)^2}{\sum_{t=1}^{n}\left[\int_{\tilde{h}=0}^{L}\tilde{h}\left[\int_{\theta=0}^{1-\tilde{h}/L}(f_X(\theta)+f_X(\theta+\tilde{h}/L))P_{min,t}(\theta,\theta+\tilde{h}/L)d\theta\right]d\tilde{h}\right]^{-1}}\\
&= \frac{1}{2nL^2}\frac{\sum_{t=1}^{n}\sum_{t'=1}^{n}\sum_{i=1}^{L}\sum_{i'=1}^{L}E_{\breve{X}_t}I(Y_i(t);U_i(t,\breve{X}_t)|X(t))E_{\breve{X}_{t'}}I(Y_{i'}(t');U_{i'}(t',\breve{X}_{t'})|X(t'))}{\sum_{t=1}^{n}\left[\int_{\tilde{h}=0}^{L}\tilde{h}\left[\int_{\theta=0}^{1-\tilde{h}/L}(f_X(\theta)+f_X(\theta+\tilde{h}/L))P_{min,t}(\theta,\theta+\tilde{h}/L)d\theta\right]d\tilde{h}\right]^{-1}}\\
&\geq \min_{t}\frac{1}{2nL^2}\frac{\sum_{t'=1}^{n}\sum_{i=1}^{L}\sum_{i'=1}^{L}E_{\breve{X}_t}I(Y_i(t);U_i(t,\breve{X}_t)|X(t))E_{\breve{X}_{t'}}I(Y_{i'}(t');U_{i'}(t',\breve{X}_{t'})|X(t'))}{\left[\int_{\tilde{h}=0}^{L}\tilde{h}\left[\int_{\theta=0}^{1-\tilde{h}/L}(f_X(\theta)+f_X(\theta+\tilde{h}/L))P_{min,t}(\theta,\theta+\tilde{h}/L)d\theta\right]d\tilde{h}\right]^{-1}}\\
&\geq \min_{t,t',i,i'}E_{\breve{X}_t}I(Y_i(t);U_i(t,\breve{X}_t)|X(t))E_{\breve{X}_{t'}}I(Y_{i'}(t');U_{i'}(t',\breve{X}_{t'})|X(t'))\\
&\qquad\times\int_{\tilde{h}=0}^{L}\frac{\tilde{h}}{2}\left[\int_{\theta=0}^{1-\tilde{h}/L}(f_X(\theta)+f_X(\theta+\tilde{h}/L))P_{min,t}(\theta,\theta+\tilde{h}/L)d\theta\right]d\tilde{h}
\end{aligned}
\tag{5.63}
$$

where Proposition A.7.1 from the Appendix is used for the last two inequalities. Since the input sequence $X(t)$ is i.i.d. over time, the minimum for the 'primed' variables and the 'unprimed' variables is the same. Therefore, one can further simplify the inequality in (5.63) as:

$$\begin{aligned} R^2 D(L,R) \geq & \left(\min_{t,i} E_{\breve{X}_t} I(Y_i(t); U_i(t, \breve{X}_t) | X(t)) \right)^2 \\ & \int_{\tilde{h}=0}^{L} \frac{\tilde{h}}{2} \left[\int_{\theta=0}^{1-\tilde{h}/L} (f_X(\theta) + f_X(\theta + \tilde{h}/L)) P_{min,t}(\theta, \theta + \tilde{h}/L) d\theta \right] d\tilde{h}. \end{aligned} \tag{5.64}$$

Further simplification gives the following

$$\begin{aligned} R^2 D(L,R) \geq & \left(\min_{t,i,\breve{X}_t} I(Y_i(t); U_i(t, \breve{X}_t) | X(t)) \right)^2 \\ & \int_{\tilde{h}=0}^{L} \frac{\tilde{h}}{2} \left[\int_{\theta=0}^{1-\tilde{h}/L} (f_X(\theta) + f_X(\theta + \tilde{h}/L)) P_{min,t}(\theta, \theta + \tilde{h}/L) d\theta \right] d\tilde{h}. \end{aligned} \tag{5.65}$$

Now as $L \to \infty$, using the Chernoff-Stein Lemma [40], the error probability $P_{min,t}(\theta, \theta+\tilde{h}/L)$ is given as $e^{-L\mathtt{C}(\theta,\theta+\tilde{h}/L)}$ where $G_\theta(\tilde{h}/L) \triangleq \mathtt{C}(\theta, \theta + \tilde{h}/L)$ is the Chernoff information between the conditional densities of y given $x = \theta$ and $x = \theta + \tilde{h}/L$. It is given by the following

$$G_\theta(\tilde{h}/L) = -\min_s \log \left(\int W_\alpha^s(y|\theta) W_\alpha^{1-s}(y|\theta + \tilde{h}/L) dy \right).$$

Since the argument of $G_\theta(\tilde{h}/L)$ is close to zero as $L \to \infty$, using the Taylor expansion of $G_\theta(\Delta)$ around zero, we get:

$$G_\theta(\Delta) = G_\theta(0) + \Delta G'_\theta(\Delta)|_{\Delta=0} + O(\Delta^2). \tag{5.66}$$

Using this expansion:

$$\begin{aligned} e^{-L\mathtt{C}(\theta,\theta+\tilde{h}/L)} &= e^{-LG_\theta(\tilde{h}/L)} && (5.67)\\ &= e^{-L(G_\theta(0)+\tilde{h}/LG'_\theta(\Delta)|_{\Delta=0}+O(L^{-2}))} \\ &= e^{-\tilde{h}G'_\theta(\Delta)|_{\Delta=0}+O(L^{-1})}, && (5.68)\end{aligned}$$

since $G_\theta(0) = \mathtt{C}(\theta,\theta) = 0$. Therefore,

$$\lim_{L\to\infty} R^2 D(L,R) \geq \lim_{L\to\infty} \left(\min_{t,i,\check{X}_t} I(Y_i(t);U_i(t,\check{X}_t)|X(t))\right)^2 \int_{\tilde{h}=0}^{L} \frac{\tilde{h}}{2}\left[\int_{\theta=0}^{1-\tilde{h}/L}(f_X(\theta)+f_X(\theta+\tilde{h}/L))e^{-\tilde{h}G'_\theta(\Delta)|_{\Delta=0}+O(L^{-1})}d\theta\right]d\tilde{h}$$

which implies

$$\beta(\alpha) = \lim_{L,R\to\infty} R^2 D(L,R) \geq \Big(\min_{U:X\to Y\to U} I(Y;U|X)\Big)^2 \int_{h=0}^{\infty} h \int_{\theta=0}^{1} f_X(\theta)e^{-hg(\theta)}d\theta dh,$$

where $g(\theta)$ is the first derivative of Chernoff information between the conditional densities (W_α) of the observation given $x=\theta$ and $x=\theta+\Delta$, evaluated at $\Delta=0$ and is given by:

$$g(\theta) = \left\{\frac{d}{d\Delta} - \left[\min_s \log\left(\int W_\alpha^s(y|\theta)W_\alpha^{1-s}(y|\theta+\Delta)dy\right)\right]\right\}_{\Delta=0}. \quad (5.69)$$

This concludes the proof of the converse.

5.5 Discussion

In this chapter, the case of unskilled local humans in a network was considered and their effect was analyzed on the estimation of a source sequence. The asymptotic behavior of the minimum achievable mean squared error distortion at the CEO was derived in the limit when the number of agents and the sum rate tend to infinity. This is the non-regular CEO problem, which addresses the practical case where multiple subordinates send quantal grades of their noisy beliefs to the CEO. When the source distribution does not satisfy the regularity conditions, we get an intermediate regime of performance between the discrete CEO problem [17] and the quadratic Gaussian CEO problem [162]. A key observation is that the rate of convergence depends on Chernoff information. The result extends the literature on the CEO problem from the traditional case of Gaussian source distribution and Gaussian channel noise to non-regular source distributions. While the proofs are similar in structure to the traditional CEO problems, they use different techniques, which can also be applied to other non-Gaussian non-regular multiterminal source coding problems. Our results indicate that one can expect a change in behavior for other multiterminal source coding problems as well, when the source follows non-Gaussian non-regular distribution.

In the next chapter, the effect of an unreliable global human decision maker fusing decisions from multiple agents is considered.

CHAPTER 6

DETECTION IN HUMAN NETWORKS: UNRELIABLE GLOBAL FUSION

6.1 Introduction

In the previous chapter, the effect of unskilled local humans on estimating a source value was explored. It considered the case when the local agents are humans and the global agent referred to as the CEO is using an optimal estimator. However, such an assumption is not valid in reality and one needs to understand how a human agent fuses data from multiple agents, to engineer systems where the GDM is also a human. This chapter studies decision fusion by humans via behavioral experiments and demonstrates differences from optimal approaches. Based on experimental results, we develop a particular bounded rationality model (cf. [54]). The implications of such a model on design of sociotechnical systems are presented by developing optimal decision fusion trees with human decision fusion components.

The remainder of the chapter is organized as follows: In Sec. 6.2, psychology experiments are described that help understand decision fusion by humans, especially in comparison to optimal fusion rules. After establishing that the existing decision fusion models of machines cannot explain the human behavior, in Sec. 6.3, Bayesian hierarchical models are developed to explain the observed behavior. In Sec. 6.4, its implications are discussed by demonstrating its effect on the design of large-scale sociotechnical systems. Concluding remarks are provided in Sec. 6.5.

6.2 Experiments and Data Analysis

Sec. 6.2.1 describes the experiments designed to study decision fusion by humans. The collected data is analyzed in Sec. 6.2.2 and the performance of humans is compared with that of optimal rules.

6.2.1 Experimental Design

To understand the decision fusion behavior in humans, experiments replicating the process of Fig. 3.2.1 were designed. Human subjects consisting of undergraduate students of Syracuse University were enrolled for this task. The experiment consisted of data collection in two stages: the first stage models local decision-making and the second stage models the data fusion aspect. The experiment was based on a memory-based task and is described as follows. Consider a target set of 100 words $\mathcal{D}$ and a distinct set of 100 distractor words $\mathcal{N}$, with $\mathcal{S} = \mathcal{D} \cup \mathcal{N}$. For the first stage, human subjects (called sources) took part in a recognition task where they first memorized $\mathcal{D}$, then performed a local decision for each $s \in \mathcal{S}$ as to whether $s \in \mathcal{D}$ or $s \in \mathcal{N}$. In the second stage, a new set of human subjects had to decide whether the word was present in the original database $\mathcal{D}$ by using decisions from the sources. These human subjects of second stage replicate the role of a GDM (Fig. 3.2.1). Note that these decision makers of second stage have no direct access to the database; their only source of information is from the sources. Local decisions from a variable number of sources (N) were presented to these subjects. This value N was either 2, 5, 10, or 20. The subjects were also presented with the sources' reliabilities and bias values. These values play the role of probability of detection and false alarm used in signal processing literature. Each dataset of the resulting dataset has the following information: word s, true hypothesis of s ($s \in \mathcal{D}$ or $s \in \mathcal{N}$), number of sources for this particular task (N), sources' decisions and reliabilities, and the fused decision reported by GDM.

6.2.2 Data Analysis

This section presents a summary of the analyzed data. First, the optimal decision fusion rule [26] is presented for comparison.

Optimal fusion rule

When the sources' reliabilities are known, optimal decision fusion is achieved by the Chair-Varshney (CV) rule [26]. Represent the "Yes/No" decisions of ith LDM as

$$u_i = \begin{cases} +1, & \text{if the decision is “Yes"}, \\ -1, & \text{if the decision is “No"}. \end{cases} \tag{6.1}$$

After receiving the N decisions $\mathbf{u} = [u_1, \ldots, u_N]$, the global decision $u_0 \in \{-1, +1\}$ is made as follows:

$$u_0 = \begin{cases} +1, & \text{if } a_0 + \sum_{i=1}^{N} a_i u_i > 0, \\ -1, & \text{otherwise}, \end{cases} \tag{6.2}$$

where $a_0 = \log \frac{P_1}{1-P_1}$,

$$a_i = \begin{cases} \log \frac{1-P_{M,i}}{P_{F,i}}, & \text{if } u_i = +1, \\ \log \frac{1-P_{F,i}}{P_{M,i}}, & \text{if } u_i = -1, \end{cases}$$

for $i = 1, \ldots, N$, and P_1 is the prior probability that the underlying hypothesis is "Yes" (+1), $P_{M,i}$, $P_{F,i}$ represent the missed detection and false alarm probabilities respectively, of the ith decision maker.

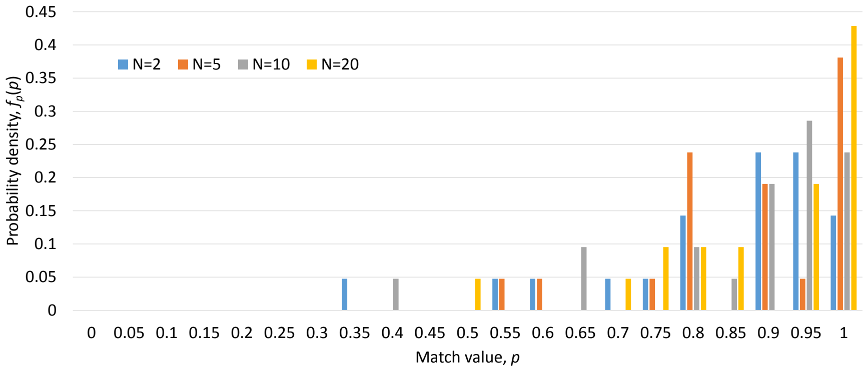


Fig. 6.1: Distribution of subjects' match value between the human decision and the CV rule's decision

How efficient are people?

To determine how efficient humans are at fusing decisions, the final decisions by 21 human subjects are compared with the decision from the Chair-Varshney rule.[1] Each human subject at the second stage typically performed 100 tasks, 25 each with $N = 2, 5, 10, 20$. The final decisions made by the humans match the optimal rule around 80–90% of the time. The closeness with the optimal fusion rule increases with an increase in N from 2 to 20. By defining the *match value* of a subject as the fraction of times his/her decision matches the decision of the CV rule with the same input data, individual participant's performance is compared with the CV rule. Although there is 80–90% match overall, a closer examination shows that the individual match value has a lot of variation across subjects. For example, when $N = 5$, while one participant had a low match value of 0.54, another participant had a high match value of 0.98. Fig. 6.1 shows the distribution of the match values between the human's decision and the CV rule's decision for different values of N. Therefore, a single decision fusion rule (such as the CV rule) cannot capture the human behavior. Next, we develop a model to represent the observed human behavior.

---

[1] Note that in our setup, $P_1 = 0.5$, implying $a_0 = 0$.

6.3 Bayesian Hierarchical Model

In this section, a Bayesian hierarchical model is developed which characterizes the human behavior at fusing multiple decisions.

6.3.1 Description of Model

The phenomenon of different match values across individual participants can be represented as a random variable following a distribution as shown in Fig. 6.1. Such a model captures the individual differences in humans while fusing multiple decisions. As mentioned before, the differences among humans can be at multiple levels: individual level, crowd level, and population level. The individual-level decision model is described below (Fig. 6.2):

- A deterministic decision v is determined using the optimal fusion rule (CV rule).
- The next step is a randomization step, where a match value p is sampled from a distribution $f_p(\cdot)$.
- The distribution $f_p(\cdot)$ is determined by fitting a model to experimental data in Fig. 6.1. The

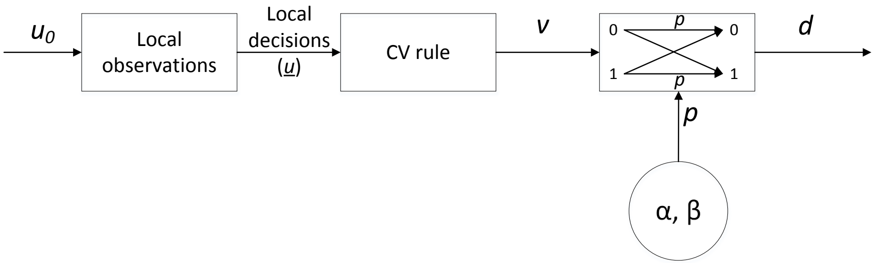


Fig. 6.2: The 2-step decision-making model where the first step determines a deterministic decision using the CV rule and the second step models the randomness of human decision-making. Here α and β are hyperparameters that capture the randomness in match value

final decision is now given by:

$$d = \begin{cases} v, & \text{with probability } p, \\ 1 - v, & \text{with probability } 1 - p. \end{cases} \tag{6.3}$$

Due to the limited number of data points, a bootstrap model is used for data fitting, where $n = 15$ data points among the total $T = 21$ data points are randomly selected for which a Beta distribution with parameters α and β are fit. This process is repeated $N_{mc} = 1000$ times. If α_j and β_j represent the parameters from the jth trial, the final parameters are decided by taking an average of these parameters. For the dataset described in Sec. 6.2, the results are shown in Fig. 6.3. An interesting observations is that the distribution $f_p(\cdot)$ shifts to the right and the mean increases with an increase in N.

Clearly the exact values of α and β are themselves dependent on the crowd considered, i.e. they depend on the number of sources, whether they are college students or online participants, the demographics of the participants, etc. This takes us to the next higher level in the model where these values of α and β, or in other words, the distribution $f_p(\cdot)$ itself is dependent on the underlying crowd chosen for the task. Different crowds would have different values of α and β. Hidden variables like demographics, motivation, etc. can affect the parameters of the randomized decision rule model discussed above. Therefore, continuing on the Bayesian modeling approach, these parameters α and β can be modeled as random variables sampled from a distribution with parameters $\mathcal{P}$ (population parameters). Population parameters govern the entire population as a whole from which different sets of crowds are sampled. This complete model can be captured by Fig. 6.4.

6.4 Optimal Design of Sociotechnical Systems

From the proposed model, it is clear that for a complete study, one has to repeat human subject experiments with different crowds, to determine the population parameters and their effect on the

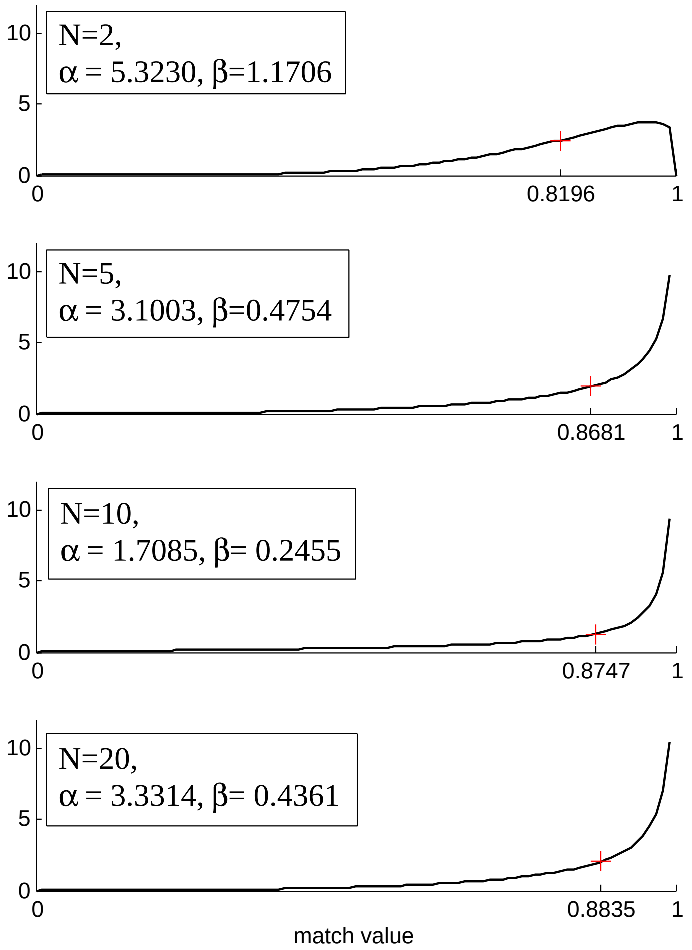


Fig. 6.3: Distribution $f_p(\cdot; \alpha, \beta)$ of match value p with parameters α and β of beta distribution, found using data fitting. The mean value is also highlighted.

crowd parameters α and β. For example, one might get different results from online participants, such as Turkers from Amazon Mechanical Turk[2], as compared to a group of college students [59]. From the experiments, an ensemble of parameters can be determined, which will help us in getting population-level insight into individual differences regarding how people fuse decisions. Such a hierarchical model can be used for understanding and designing larger signal processing systems that have a human decision fusion component such as distributed detection systems [94,154] where each agent is not a single cognitive agent, but rather a human-based decision fusion system (see Fig. 6.5). Also, cognitive agents (humans) in such systems may be drawn from a specialized sub-population.

---

[2] https://www.mturk.com/

Next, we consider designing sociotechnical systems with machines and with humans, as modeled through our hierarchical Bayesian framework. Consider a system like Fig. 6.5 where multiple levels of decision makers are present in the system with human decision makers fusing data from multiple subordinate agents (humans or sensors) before sending their fused observations to a final fusion center. If these last level agents were sensors/machines rather than humans, one can use the optimal fusion rule to fuse the data [26]. Note that this optimal fusion rule weighs the decisions with their 'reliabilities' which are deterministically known. However, when the final fusion center receives data from humans, one needs to use the Bayesian hierarchical model of human decision fusers to design the fusion rule at the FC. [3]

Considering the Bayesian formulation, the optimal fusion rule at the FC is developed by adopting a methodology similar to [26]. Let the phenomenon of interest be a binary hypothesis testing problem with prior probabilities $P(H_0) = P_0$ and $P(H_1) = P_1 = 1 - P_0$. Assume that the FC receives decisions from N_c human decision fusion components. We represent their decisions by $d_i \in \{-1, +1\}$, where $d_i = -1(+1)$, if the ith component's decision is $H_0(H_1)$. The FC makes the final decision $d_0 = f(d_1, \ldots, d_{N_c})$ using the N_c decisions based on the fusion rule $f(\cdot)$. The goal is to design the optimal fusion rule $f(\cdot)$ based on the hierarchical decision-making model of the components as discussed above (see Fig. 6.4).

The optimal decision rule that minimizes the probability of error at the FC is given by the following likelihood ratio test

$$\frac{P(d_1, \ldots, d_{N_c}|H_1)}{P(d_1, \ldots, d_{N_c}|H_0)} \underset{H_0}{\overset{H_1}{\gtrless}} \frac{P_0}{P_1}, \tag{6.4}$$

---

[3] Note there are two kinds of hierarchies considered herein: the Bayesian hierarchy for human modeling and tree hierarchy of decision-making.

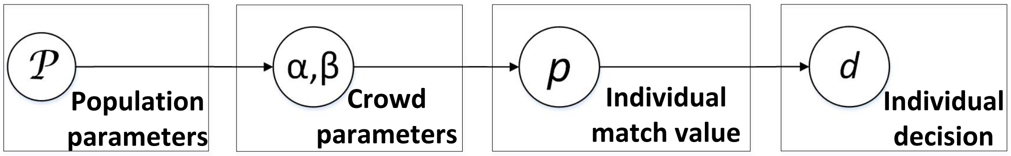


Fig. 6.4: Bayesian hierarchical model of decision fusion by humans

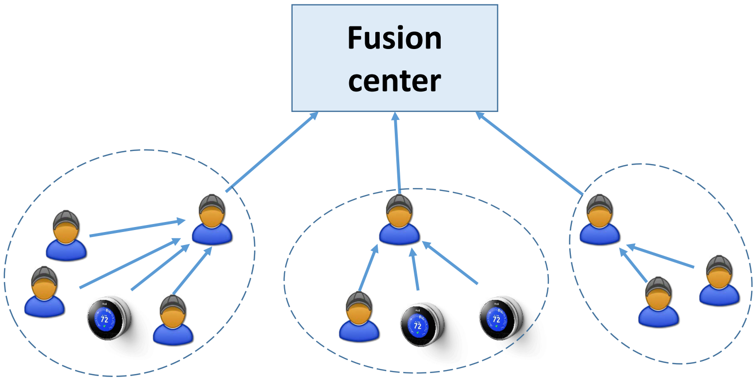


Fig. 6.5: Hierarchical system consisting of human decision fusion components

or equivalently,

$$\log \frac{P(H_1|d_1,\ldots,d_{N_c})}{P(H_0|d_1,\ldots,d_{N_c})} \underset{H_0}{\overset{H_1}{\gtrless}} 0. \tag{6.5}$$

This optimal fusion rule can be written as

$$\log \frac{P_1}{P_0} + \sum_{\mathcal{S}_\oplus} \log \frac{P(d_i=+1|H_1)}{P(d_i=+1|H_0)} + \sum_{\mathcal{S}_\ominus} \log \frac{P(d_i=-1|H_1)}{P(d_i=-1|H_0)} \underset{H_0}{\overset{H_1}{\gtrless}} 0, \tag{6.6}$$

where $\mathcal{S}_\oplus(\mathcal{S}_\ominus)$ is the set of all components that reported a decision $d_i = +1(-1)$.

The terms in (6.6) can be further simplified as

$$\begin{aligned}
&P(d_i=+1|H_1)\\
&= P(d_i=+1, d_{i,CV}=+1|H_1) + P(d_i=+1, d_{i,CV}=-1|H_1)\\
&= P(d_i=+1|d_{i,CV}=+1)P(d_{i,CV}=+1|H_1)\\
&+ P(d_i=+1|d_{i,CV}=-1)P(d_{i,CV}=-1|H_1)\\
&= p_iP_{d,i} + (1-p_i)(1-P_{d,i})\\
&= 1 - p_i - P_{d,i} + 2p_iP_{d,i}
\end{aligned}$$

where $d_{i,CV} \in \{-1,+1\}$ is the decision that the ith human fusion center would make using the optimal CV rule, p_i is the match value of the ith human, and $P_{d,i} \triangleq P(d_{i,CV}=+1|H_1)$. Similarly, the expressions for $P(d_i=+1|H_0)$, $P(d_i=-1|H_1)$, and $P(d_i=-1|H_0)$ can be derived as a function of $P_{f,i} \triangleq P(d_{i,CV} = +1|H_0)$.

$$P(d_i = +1|H_0) = 1 - p_i - P_{f,i} + 2p_iP_{f,i}, \tag{6.7}$$

$$P(d_i = -1|H_1) = p_i + P_{d,i} - 2p_iP_{d,i}, \tag{6.8}$$

and

$$P(d_i = -1|H_0) = p_i + P_{f,i} - 2p_iP_{f,i}, \tag{6.9}$$

This simplifies the optimal fusion rule (6.6) as

$$\log\frac{P_1}{P_0} + \sum_{\mathcal{S}_\oplus} \log\frac{1 - p_i - P_{d,i} + 2p_iP_{d,i}}{1 - p_i - P_{f,i} + 2p_iP_{f,i}} + \sum_{\mathcal{S}_\ominus} \log\frac{p_i + P_{d,i} - 2p_iP_{d,i}}{p_i + P_{f,i} - 2p_iP_{f,i}} \underset{H_0}{\overset{H_1}{\gtrless}} 0. \tag{6.10}$$

Note that the above expression requires the knowledge of individual match values. When this knowledge is not available, but the crowd parameters α and β are known (refer to Fig. 6.4), it is not very difficult to see that the only change in the optimal fusion rule will be to replace p_i with $E[p_i] = \frac{\alpha}{\alpha+\beta}$. Therefore, when all the decision fusion components are identical (same number of sources, identically distributed sources, etc.), then the optimal fusion rule becomes a K out of N rule. The optimal K^* can be derived as follows. Let the number of components with final decision $d_i = +1$ be K, then (6.10) becomes (after replacing p_i with $E[p_i]$)

$$\log\frac{P_1}{P_0} + K\log\frac{1 - \frac{\alpha}{\alpha+\beta} - P_d + 2\frac{\alpha}{\alpha+\beta}P_d}{1 - \frac{\alpha}{\alpha+\beta} - P_f + 2\frac{\alpha}{\alpha+\beta}iP_f} + (N_c - K)\log\frac{\frac{\alpha}{\alpha+\beta} + P_d - 2\frac{\alpha}{\alpha+\beta}P_d}{\frac{\alpha}{\alpha+\beta} + P_f - 2\frac{\alpha}{\alpha+\beta}P_f} \underset{H_0}{\overset{H_1}{\gtrless}} 0 \tag{6.11}$$

or in other words

$$K \underset{H_0}{\overset{H_1}{\gtrless}} \frac{\log\frac{P_0}{P_1} - N_c\log a^*_\ominus}{\log\frac{a^*_\oplus}{a^*_\ominus}}. \tag{6.12}$$

where

$$a^*_\oplus = \frac{1 - \frac{\alpha}{\alpha+\beta} - P_d + 2\frac{\alpha}{\alpha+\beta}P_d}{1 - \frac{\alpha}{\alpha+\beta} - P_f + 2\frac{\alpha}{\alpha+\beta}P_f} = \frac{\beta + (\alpha-\beta)P_d}{\beta + (\alpha-\beta)P_f}$$

and

$$a^*_{\ominus} = \frac{\frac{\alpha}{\alpha+\beta} + P_d - 2\frac{\alpha}{\alpha+\beta}P_d}{\frac{\alpha}{\alpha+\beta} + P_f - 2\frac{\alpha}{\alpha+\beta}P_f} = \frac{\alpha + (\beta-\alpha)P_d}{\alpha + (\beta-\alpha)P_f}.$$

This gives a K out of N rule with optimal K^* given by

$$K^* = \left\lceil \frac{\log\frac{P_0}{P_1} - N_c \log a^*_{\ominus}}{\log\frac{a^*_{\oplus}}{a^*_{\ominus}}} \right\rceil. \tag{6.13}$$

If these data fusion components of Fig. 6.5 are from different crowds, one can go higher in the Bayesian hierarchical model and use the population parameters to determine the optimal fusion rule. Also, any machines using CV rules in the penultimate level of the hierarchical sociotechnical system can be regarded as a human agent with a perfect match value of 1. Such a generality can help us in potentially constructing arbitrary-depth trees of sociotechnical decision-making, where humans are modeled and the machines are optimized.

In the following, the benefit associated with the Bayesian hierarchical model is characterized. Consider the case when such a model of human decision fusion is ignored, then the optimal K^*_{sen} for the K out of N rule is given by

$$K^*_{sen} = \left\lceil \frac{\log\frac{P_0}{P_1} - N_c \log\frac{1-P_d}{1-P_f}}{\log\frac{P_d(1-P_f)}{P_f(1-P_d)}} \right\rceil. \tag{6.14}$$

The error probability for fixed K is

$$\begin{aligned} P_e(K) = P_0 \sum_{i=K}^{N_c} \binom{N_c}{i} \left(\frac{\beta + (\alpha-\beta)P_f}{\alpha+\beta}\right)^i \left(\frac{\alpha - (\alpha-\beta)P_f}{\alpha+\beta}\right)^{N_c-i} \\ + P_1 \sum_{i=0}^{K-1} \binom{N_c}{i} \left(\frac{\beta + (\alpha-\beta)P_d}{\alpha+\beta}\right)^i \left(\frac{\alpha - (\alpha-\beta)P_d}{\alpha+\beta}\right)^{N_c-i}. \end{aligned} \tag{6.15}$$

Therefore, the performance loss by ignoring the effect of humans in the system is due to the mismatched K value and is given by (6.16).

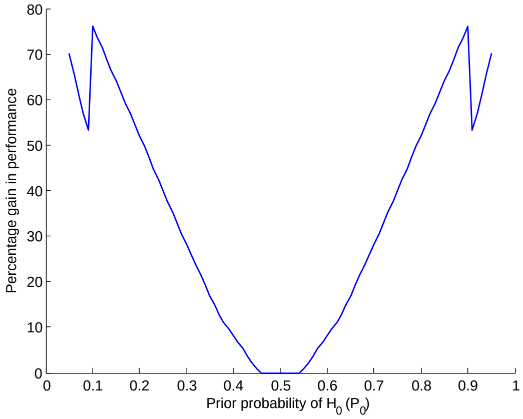


Fig. 6.6: Percentage improvement in system performance by using the Bayesian hierarchical model for system design with varying prior probability

$$\Delta P_e = \begin{cases} \sum_{i=K^*_{sen}}^{K^*-1} \binom{N_c}{i} \left[P_0 \left(\frac{\beta+(\alpha-\beta)P_f}{\alpha+\beta}\right)^i \left(\frac{\alpha-(\alpha-\beta)P_f}{\alpha+\beta}\right)^{N_c-i} - P_1 \left(\frac{\beta+(\alpha-\beta)P_d}{\alpha+\beta}\right)^i \left(\frac{\alpha-(\alpha-\beta)P_d}{\alpha+\beta}\right)^{N_c-i} \right], & \text{if } K^* > K^*_{sen}, \\ \sum_{i=K^*}^{K^*_{sen}-1} \binom{N_c}{i} \left[P_1 \left(\frac{\beta+(\alpha-\beta)P_d}{\alpha+\beta}\right)^i \left(\frac{\alpha-(\alpha-\beta)P_d}{\alpha+\beta}\right)^{N_c-i} - P_0 \left(\frac{\beta+(\alpha-\beta)P_f}{\alpha+\beta}\right)^i \left(\frac{\alpha-(\alpha-\beta)P_f}{\alpha+\beta}\right)^{N_c-i} \right], & \text{if } K^* < K^*_{sen} \end{cases} \tag{6.16}$$

In Fig. 6.6, the performance gain by using the Bayesian hierarchical model is plotted against different values of prior probability. The parameters used are $N_c = 5$, $P_d = 0.9$, $P_f = 0.1$, $\alpha = 5$, and $\beta = 3$. As can be observed, by utilizing the knowledge of human decision fusion components in the system during system design, one can improve the performance by around 35% on an average.

The sudden jump in performance gain around priors $P_0 = 0.1$ and $P_0 = 0.9$ is due to the chosen values of P_d and P_f and can be analytically determined using the expressions in (6.13) and (6.14). Also, note that the region around $P_0 = 0.5$ for which there is no performance improvement is due to the situation when the term dependent on the prior dominates the other terms in the expressions of K^* and K^*_{sen}, thereby resulting in equal values of K^* and K^*_{sen}. The width of this region where there is no performance gain depends on the values of α and β as we can see in Fig. 6.7, as $P_0 = 0.3$ is outside this region for $\beta \geq 1.5$ while it is within this region for $\beta < 1.5$. Similar observations can be made for different values of priors.

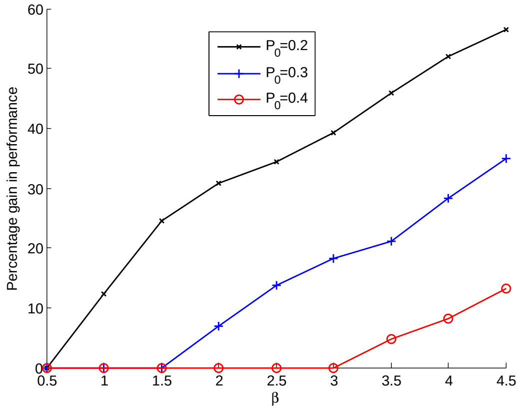


Fig. 6.7: Percentage improvement in system performance by using the Bayesian hierarchical model for system design with varying values of β and $\alpha = 0.5$

6.5 Discussion

In this chapter, the decision fusion problem has been considered. It was first observed for a system involving humans that a deterministic optimal fusion rule, such as the Chair-Varshney rule, does not characterize the human behavior, since data fusion by humans is not deterministic in nature. For a given set of data, the optimal deterministic rule gives the same output at any time instant. On the other hand, the output changes for different humans and in some cases, for the same human at different time instant, as pointed by Payne and Bettman in [107]. This suggests the use of a randomized decision rule, which was the focus of the next part of the chapter. Hierarchical models have been developed which characterize this behavior. The effect of such models on the design of larger human-machine systems has been demonstrated.

Having analyzed the effect of unreliable humans in this chapter and the previous one, in the next chapter, human networks with such unskilled humans are considered and a coding theory based scheme to ensure reliable inference from them is developed.

CHAPTER 7

CLASSIFICATION IN HUMAN NETWORKS: RELIABLE INFERENCE

7.1 Introduction

In the previous chapters, the effect of unskilled humans in the network was analyzed and it was shown that the performance can be suboptimal due to their presence. In this chapter, the case when the local agents are humans but the global agent is a machine is considered and schemes are designed that can use such unskilled humans and still yield reasonable performance at the FC. This setup is that of crowdsourcing where workers often find microtasks tedious and due to lack of motivation fail to generate high-quality work [112]. It is, therefore, important to design crowdsourcing systems with sufficient incentives for workers [83]. The most common incentive for workers is monetary reward, but in [151], it has been found that intrinsic factors such as the challenge associated with the task was a stronger motivation for crowd workers than extrinsic factors such as rewards. Unfortunately, it has been reported that increasing financial incentives increases the number of tasks which the workers take part in but not the per task quality [96]. Recent research, however, suggests that making workers' rewards co-dependent on each other can significantly increase the quality of their work. This suggests the potency of a *peer-dependent reward scheme* for quality control [61]. In a *teamwork-based scheme*, paired workers are rewarded based on their average work whereas in a *competition-based* scheme, the paired worker who performs the best gets the entire reward. Herein, we develop a mathematical framework to evaluate the effect of such pairings among crowd workers.

Another interesting phenomenon in crowdsourcing is *dependence* of observations among crowd workers [111]. Crowds may share common sources of information, leading to dependent observations among the crowd workers performing the task. Common sources of information have oft-been implicated in the publishing and spread of false information across the internet, e.g. the premature Steve Jobs obituary, the second bankruptcy of United Airlines, and the creation of black holes by operating the Large Hadron Collider [19]. A graphical model may be used to characterize such dependence among crowd workers [111].

In typical networked sensing systems, heterogeneous sensors are used to observe a phenomenon and to collaboratively make inferences (detection, classification, or estimation), where algorithms at the sensors or at a fusion center are derived to operate optimally or near-optimally. In large sensor networks consisting of inexpensive battery-powered sensors with limited capabilities, one key issue has been to maintain inference quality in the presence of faulty sensors or communication errors. Recently an innovative marriage of concepts from coding theory and distributed inference has been proposed [166, 170], where the goal is to jointly maximize classification performance and system fault tolerance by jointly designing codes and decision rules at the sensors. In the present work, we apply distributed inference codes [166] to crowdsourcing tasks like classification. This is consistent with popular uses of crowdsourcing microtask platforms such as Amazon Mechanical Turk.

Since quality control is a central concern for crowdsourcing [62], previous work considered numerical analysis methods [55] and binary classification tasks [70, 71]. In [70, 71], the authors considered the problem of task allocation in a crowdsourcing system; an iterative algorithm based on belief propagation was proposed for inferring the final answer from the workers' responses. This algorithm was shown to perform as well as the best possible algorithm. They also provided numerical results for large system size and showed that their approach outperforms the majority-based approach. Extensions to a multi-class labeling task [72] provided an algorithm to obtain the best tradeoff between reliability and redundancy. This algorithm was based on low-rank approximation of weighted adjacency matrices for random regular bipartite graphs used for task allocation.

In this chapter, a coding theory based scheme is proposed which has two-fold benefits. The coding-based scheme helps us in designing easy-to-answer binary questions for the humans that improves the performance of individual agents. The second benefit is on the decoding side where the error-correcting code can tolerate a few errors from these agents and, therefore, make reliable inferences. The remainder of the chapter is organized as follows: In Sec. 7.2, a mathematical model of the crowdsourcing problem is developed and the coding-based approach is proposed. In Sec. 7.3, a crowdsourcing system with independent workers is considered. The peer-dependent reward scheme is introduced in Sec. 7.4 and the system is further generalized in Sec. 7.5 by allowing dependence among crowd worker observations. For each of these models, in turn, misclassification performance expressions for both coding- and majority-based approaches are derived. Examples demonstrate that systems with good codes outperform systems that use majority voting. Experimental results using real data from Amazon Mechanical Turk are also provided wherever applicable. Concluding remarks are provided in Sec. 7.6.

7.2 Coding for Crowdsourcing

In this section, the basic concept of using error-correcting codes to achieve reliable classification in a crowdsourcing system is discussed.

7.2.1 Reliable Classification using Crowds

First, we describe how the DCFECC approach discussed in Sec. 2.4 can be used in crowdsourcing systems to design the questions to be posed to the crowd workers. As an example, consider an image to be classified into one of M fine-grained categories. Since object classification is often difficult for machine vision algorithms, human workers may be used for this task. In a typical crowdsourcing microtask platform, a task manager creates simple tasks for the workers to complete, and the results are combined to produce the final result. Due to the low pay of workers and the difficulty of tasks, individual results may be unreliable. Furthermore, workers may not be qualified to make fine-grained M-ary distinctions, but rather can only answer easier questions. Therefore, in the proposed approach, codes are used to design microtasks and decoding is performed to aggregate responses reliably.

Consider the task of classifying a dog image into one of four breeds: Pekingese, Mastiff, Maltese, or Saluki. Since workers may not be canine experts, they may not be able to directly classify and so we should ask simpler questions. For example, the binary question of whether a dog has a snub nose or a long nose differentiates between {Pekingese, Mastiff} and {Maltese, Saluki}, whereas the binary question of whether the dog is small or large differentiates between {Pekingese, Maltese} and {Mastiff, Saluki}. Using a code matrix, we now show how to design binary questions for crowd workers that allow the task manager to reliably infer correct classification even with unreliable workers.

As part of modeling, assume that worker j decides the true class (local decision y_j) with probability p_j and makes the wrong classification with uniform probability:

$$p(y_j|H_m) = \begin{cases} p_j & \text{if } y_j = m \\ \frac{1-p_j}{M-1} & \text{otherwise,} \end{cases} \tag{7.1}$$

Note that, the uniform noise model here can be regarded as the "worst case" in terms of information. In such a case, this would relate to an upper bound on the performance of the system. For every worker j, let a_j be the corresponding column of $\mathbf{A}$ and recall hypothesis $H_l \in \{H_0, H_1, \cdots, H_{M-1}\}$ is associated with row l in $\mathbf{A}$. The local workers send a binary answer u_j based on decision y_j and column a_j. An illustrative example is shown in Fig. 7.1 for the dog breed classification task above. Let the columns corresponding to the ith and jth workers be

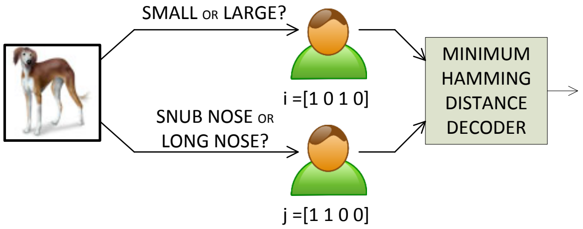


Fig. 7.1: A schematic diagram showing binary questions posed to workers and the decoding rule used by the task manager

$a_i = [1010]'$ and $a_j = [1100]'$ respectively. The ith worker is asked: "Is the dog small or large?" since the worker is to differentiate between the first (Pekingese) or third (Maltese) breed and the others. The jth worker is asked: "Does the dog have a snub nose or a long nose?" since the worker is to differentiate between the first two breeds (Pekingese, Mastiff) and the others. These questions can be designed using taxonomy and dichotomous keys [121]. Knowing that Pekingese and Maltese are small dogs while the other two breeds are large, we can design the appropriate question as "Is the dog small or large?" for ith worker whose corresponding column is $a_i = [1010]'$. The task manager makes the final classification as the hypothesis corresponding to the codeword (row) that is closest in Hamming distance to the received vector of decisions.

7.2.2 Unreliable Workers

Although distributed classification in sensor networks and in crowdsourcing are structurally similar, an important difference is the anonymity of crowds. Since crowd workers are anonymous, one cannot identify the specific reliability of any specific worker as could be done for a sensor. Hence, each worker j in the crowd is assumed to have an associated reliability p_j, drawn from a common distribution that characterizes the crowd. Herein, three different crowd models that generate crowd reliabilities are considered: individual and independent crowd workers; crowd workers governed by peer-dependent reward schemes; and crowd workers with common sources of information. In the following sections, each of these models is analyzed to evaluate the proposed coding-based scheme.

7.3 Crowdsourcing System with Individual Crowd Workers

In this section, the basic crowdsourcing system is analyzed where independent crowd workers perform the task individually and are rewarded based on their decision only.

7.3.1 Model

The system consisting of individual and independent workers can be modeled as one where the workers' reliabilities are drawn i.i.d. from a specific distribution. Two crowd reliability models namely a spammer-hammer model and a beta model are considered herein. In a spammer-hammer model, the crowd consists of two kinds of workers: spammers and hammers. Spammers are unreliable workers that make a decision at random whereas hammers are reliable workers that make a decision with high reliability. The quality of the crowd, Q, is governed by the fraction of hammers. In a beta model, the reliabilities of workers are drawn from a beta distribution with parameters α and β.

7.3.2 Performance Characterization

Having defined a coding-based approach to reliable crowdsourcing, its performance is determined in terms of average misclassification probability for classification under minimum Hamming distance decoding. Suppose N workers take part in an M-ary classification task. Let **p** denote the reliabilities of these workers, such that p_j for $j = 1, \ldots, N$ are i.i.d. random variables with mean μ. Define this to be an (N, M, μ) crowdsourcing system.

Proposition 7.3.1. *Consider an* (N, M, μ) *crowdsourcing system. The expected misclassification* *probability using code matrix* $\mathbf{A}$ *is:*

$$P_e(\mu) \quad = \quad \frac{1}{M}\sum_{\boldsymbol{i},l}\prod_{j=1}^{N}\left[\left(\mu a_{lj}+\frac{(1-\mu)}{(M-1)}\sum_{k\neq l}a_{kj}\right)(2i_j \quad - \quad 1) \quad + \quad (1 \quad - \quad i_j)\right]C_{\boldsymbol{i}}^{l}, \quad (7.2)$$

where $\boldsymbol{i} = [i_1, \cdots, i_N] \in \{0,1\}^N$ *is the received codeword and* $C_{\boldsymbol{i}}^l$ *is the cost associated with a global decision* H_l *when the received vector is* $\boldsymbol{i}$*. This cost is:*

$$C_{\boldsymbol{i}}^{l} = \begin{cases} 1-\frac{1}{\varrho} & \textit{if } \boldsymbol{i} \textit{ is in decision region of } H_l \\ 1 & \textit{otherwise.} \end{cases} \quad (7.3)$$

where ϱ *is the number of decision regions*[1] $\boldsymbol{i}$ *belongs to;* ϱ *can be greater than one when there is a tie at the task-manager and the tie-breaking rule is to choose one of them randomly.*

Proof. Let $P_{e,\mathbf{p}}$ denote the misclassification probability given the reliabilities of the N workers. Then, if u_j denotes the bit sent by the worker j and the global decision is made using the Hamming distance criterion:

$$P_{e,\mathbf{p}} = \frac{1}{M}\sum_{\mathbf{i},l}P(\mathbf{u}=\mathbf{i}|H_l)C_{\mathbf{i}}^{l}. \quad (7.4)$$

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

$$\begin{aligned} P(u_j = i_j|H_l) &= i_j P(u_j = 1|H_l) + (1-i_j)P(u_j = 0|H_l) \\ &= (1-i_j) + (2i_j - 1)P(u_j = 1|H_l) \\ &= (1-i_j) + (2i_j-1)\sum_{k=1}^{M} a_{kj}P(y_j = k|H_l) \\ &= (1-i_j) + \left(p_j a_{lj} + \frac{(1-p_j)}{(M-1)}\sum_{k\neq l}a_{kj}\right)(2i_j - 1) \end{aligned}$$

where y_j is the local decision made by worker j. Since reliabilities p_j are i.i.d. with mean μ, the desired result follows. □

---

[1] For each H_l, the set of **i** for which the decision H_l is taken is called the decision region of H_l.

Performance Bound

Yao et al. provide performance analysis for the distributed M-ary classification fusion system with minimum Hamming distance fusion [170]. Their result is stated here without proof. This result can be used in the context of distributed M-ary classification using N workers with reliabilities $\{p_j\}_{j=1}^N$ and a code matrix $\mathbf{A}$ for coding-based classification.

Proposition 7.3.2 ([170])**.** *Let P_e be the probability of minimum Hamming distance fusion misclassification error given as*

$$P_e \triangleq \frac{1}{M}\sum_{i=1}^{M-1} P(\textit{fusion decision} \neq H_i|H_i). \tag{7.5}$$

If for every $l \neq i$

$$\sum_{\{j\in[1,\cdots,N]:a_{lj}\neq a_{ij}\}} E[z_{i,j}] = \sum_{j=1}^{N}(a_{lj}\oplus a_{ij})(2q_{i,j}-1) < 0, \tag{7.6}$$

where $0 \leq l, i \leq M-1$, $z_{i,j} \triangleq 2(u_j \oplus a_{ij}) - 1$, $\oplus$ represents the 'xor' operation and $q_{i,j} \triangleq P\{z_{i,j} = 1|H_i\}$, then

$$P_e \quad \leq \quad \frac{1}{M}\sum_{i=0}^{M-1}\sum_{0\leq l\leq M-1, l\neq i}\inf_{\theta\geq 0}\exp\left\{\sum_{j=1} N\log\left(q_{i,j}e^{\theta}+(1-q_{i,j})e^{-\theta}\right)^{a_{lj}\oplus a_{ij}}\right\}. \tag{7.7}$$

The proof of the proposition follows from large deviations theory [170]. In crowdsourcing, the probabilities $q_{i,j} = P\{u_j \neq a_{ij}\}$ can be easily computed as:

$$q_{i,j} \quad = \quad \sum_{l=0}^{M-1}(a_{ij}\oplus a_{lj})h_{l|i}^{(j)}, \tag{7.8}$$

where $h_{l|i}^{(j)}$ is the probability that worker j decides H_l when the true hypothesis is H_i and are given by

$$h_{l|i}^{(j)} = \begin{cases} p_j, & i = l \\ \frac{1-p_j}{M-1}, & i \neq l. \end{cases} \tag{7.9}$$

7.3.3 Majority Voting

A traditional approach in crowdsourcing has been to use a majority vote to combine local decisions; we also derive its performance for purposes of comparison. For M-ary classification, each worker's local decision is modeled as $\log_2 M$-bit valued, but since workers only answer binary questions, the N workers are split into $\log_2 M$ groups with each group sending information regarding a single bit. For example, consider the dog breed classification task of Sec. 7.2.1 which has $M = 4$ classes. Let us represent the classes by 2-bit numbers as follows: Pekingese is represented as '00', Mastiff as '01', Maltese as '10', and Saluki as '11'. The N crowd workers are split into 2 groups. In traditional majority vote, since the workers are asked M-ary questions, each worker first identifies his/her answer. After identifying his/her class, the first group members send the first bit corresponding to their decisions, while the second group members send the second bit of their decisions. The task manager uses a majority rule to decide each of the $\log_2 M$ bits separately and concatenates to make the final classification. Suppose N is divisible by $\log_2 M$.

Proposition 7.3.3. *Consider an (N, M, μ) crowdsourcing system. The expected misclassification probability using majority rule is:*

$$P_e(\mu) = 1 - \frac{1}{M}\left[1 + S_{\tilde{N},(1-q)}\left(\frac{\tilde{N}}{2}\right) - S_{\tilde{N},q}\left(\frac{\tilde{N}}{2}\right)\right]^{\log_2 M}, \tag{7.10}$$

where $\tilde{N} = \frac{N}{\log_2 M}$, $q = \frac{M(1-\mu)}{2(M-1)}$, and $S_{N,p}(\cdot)$ is the survival function (complementary cumulative distribution function) of the binomial random variable $\mathcal{B}(N, p)$.

Proof. In a majority-based approach, $\tilde{N} = \frac{N}{\log_2 M}$ workers send information regarding the ith bit of their local decision, $i = 1, \ldots, \log_2 M$. For a correct global decision, all bits have to be correct. Consider the ith bit and let $P_{c,\mathbf{p}}^{i}$ be the probability of the ith bit being correct given the reliabilities of the $\tilde{N}$ workers sending this bit. Then,

$$P^i_{c,\mathbf{p}} = \frac{P_d + 1 - P_f}{2}, \tag{7.11}$$

where P_d is the probability of detecting the ith bit as '1' when the true bit is '1' and P_f is the probability of detecting the ith bit as '1' when the true bit is '0'. Note that '0' and '1' are equiprobable since all hypotheses are equiprobable. Under majority rule for this ith bit,

$$P_d = \sum_{j=\lfloor \frac{\tilde{N}}{2}+1 \rfloor}^{\tilde{N}} \sum_{\forall G_j} \prod_{k \in G_j} \left(1 - \frac{M(1-p_k)}{2(M-1)}\right) \prod_{k \notin G_j} \frac{M(1-p_k)}{2(M-1)},$$

where G_j is a set of j out of $\tilde{N}$ workers who send bit value '1' and $\frac{M(1-p_k)}{2(M-1)}$ is the probability of the kth worker making a wrong decision for the ith bit. Similarly,

$$P_f = \sum_{j=\lfloor \frac{\tilde{N}}{2}+1 \rfloor}^{\tilde{N}} \sum_{\forall G_j} \prod_{k \in G_j} \frac{M(1-p_k)}{2(M-1)} \prod_{k \notin G_j} \left(1 - \frac{M(1-p_k)}{2(M-1)}\right).$$

Now the overall probability of correct decision is given by $P_{c,\mathbf{p}} = \prod_{i=1}^{\log_2 M} P^i_{c,\mathbf{p}}$. Since reliabilities are i.i.d., the expected probability of correct decision P_c is:

$$P_c = \prod_{i=1}^{\log_2 M} E[P^i_{c,\mathbf{p}}], \tag{7.12}$$

where expectation is with respect to $\mathbf{p}$. Since reliabilities are i.i.d.:

$$E[P_d] = \sum_{j=\lfloor \frac{\tilde{N}}{2}+1 \rfloor}^{\tilde{N}} \binom{\tilde{N}}{j} (1-q)^j q^{(\tilde{N}-j)} = S_{\tilde{N},(1-q)}\left(\frac{\tilde{N}}{2}\right), \tag{7.13}$$

$$E[P_f] = \sum_{j=\lfloor \frac{\tilde{N}}{2}+1 \rfloor}^{\tilde{N}} \binom{\tilde{N}}{j} q^j (1-q)^{(\tilde{N}-j)} = S_{\tilde{N},q}\left(\frac{\tilde{N}}{2}\right). \tag{7.14}$$

Using (7.11), (7.12), (7.13), and (7.14), we get the desired result. □

7.3.4 Performance Evaluation

The expressions derived in the previous subsection help us in understanding the behavior of crowdsourcing systems. One can define an ordering principle for the quality of crowds in terms of the quality of their distributed inference performance. This is a valuable concept since it provides us a tool to evaluate a given crowd. Such a valuation could be used by the task manager to pick the appropriate crowd for the task based on the performance requirements. For example, if the task manager is interested in constraining the misclassification probability of his/her task to ϵ while simultaneously minimizing the required crowd size, the above expressions can be used to choose the appropriate crowd.

Theorem 7.3.4 (Ordering of Crowds)**.** *Consider crowdsourcing systems involving crowd $\mathcal{C}(\mu)$ of workers with i.i.d. reliabilities with mean μ. Crowd $\mathcal{C}(\mu)$ performs better than crowd $\mathcal{C}(\mu')$ for classification if and only if $\mu > \mu'$.*

Proof. As can be observed from Props. 7.3.1 and 7.3.3, the average misclassification probabilities depend only on the mean of the reliabilities of the crowd. Therefore, it follows that crowd $\mathcal{C}(\mu)$ of workers with i.i.d. reliabilities with mean μ performs better for classification than crowd $\mathcal{C}(\mu')$ of workers with i.i.d. reliabilities with mean μ' as $\mu > \mu'$. □

Since the performance criterion is average misclassification probability, this can be regarded as a weak criterion of crowd-ordering in the mean sense. Thus, with this crowd-ordering, better crowds yield better performance in terms of average misclassification probability. Indeed, misclassification probability decreases with better quality crowds. In this chapter, the term reliability has been used to describe the individual worker's reliability while the term quality is a description of the total reliability of a given crowd (a function of mean μ of worker reliabilities). For example, for the spammer-hammer model, quality of the crowd is a function of the number of hammers in the crowd, while the individual crowd workers have different reliabilities depending on whether the worker is a spammer or a hammer.

Proposition 7.3.5. *Average misclassification probability reduces with increasing quality of the crowd.*

Proof. Observe from Props. 7.3.1 and 7.3.3 for coding- and majority-based approaches, respectively, that the average misclassification probability is a monotonically decreasing function of the mean of reliabilities of the crowd (μ). This value μ serves as a quality parameter of the crowd and, therefore, average misclassification probability reduces with increasing quality of the crowd. □

To get more insight, a crowdsourcing system with coding is simulated as follows: $N = 10$ workers take part in a classification task with $M = 4$ equiprobable classes. A good code matrix $\mathbf{A}$ is found by simulated annealing [166]:

$$\mathbf{A} = [5, 12, 3, 10, 12, 9, 9, 10, 9, 12]. \tag{7.15}$$

Here and in the sequel, code matrices are represented as a vector of M bit integers. Each integer r_j represents a column of the code matrix $\mathbf{A}$ and can be expressed as $r_j = \sum_{l=0}^{M-1} a_{lj} \times 2^l$. For example, the integer 5 in column 1 of $\mathbf{A}$ represents $a_{01} = 1$, $a_{11} = 0$, $a_{21} = 1$ and $a_{31} = 0$.

Consider the setting where all the workers have the same reliability $p_j = p$. Fig. 7.2 shows the probability of misclassification as a function of p. As is apparent, the probability of misclassification reduces with reliability and approaches 0 as $p \to 1$, as expected.

Now the performance of the coding-based approach is compared to the majority-based approach. Fig. 7.3 shows misclassification probability as a function of crowd quality for $N = 10$ workers taking part in an $(M = 4)$-ary classification task. The spammer-hammer model, where spammers have reliability $p = 1/M$ and hammers have reliability $p = 1$, is used. The figure shows a slight improvement in performance over majority vote when code matrix (7.15) is used.

Now consider a larger system with increased M and N. A good code matrix $\mathbf{A}$ for $N = 15$ and $M = 8$ is found by cyclic column replacement:

$$\mathbf{A} = [150, 150, 90, 240, 240, 153, 102, 204, 204, 204, 170, 170, 170, 170, 170]. \tag{7.16}$$

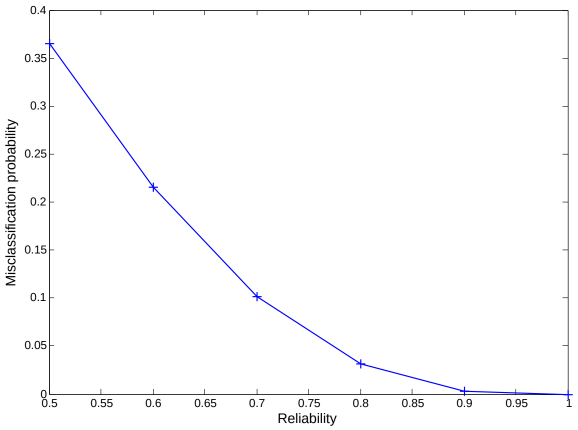


Fig. 7.2: Coding-based crowdsourcing system misclassification probability as a function of worker reliability

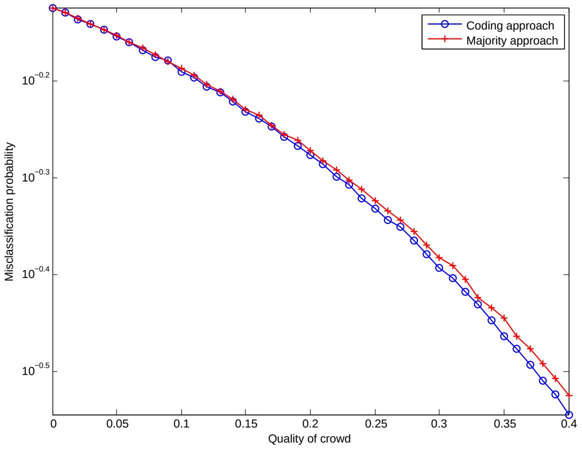


Fig. 7.3: Misclassification probability as a function of crowd quality using coding- and majority-based approaches with the spammer-hammer model, $(M = 4, N = 10)$.

The code matrix for the system with $N = 90$ and $M = 8$ is formed sub-optimally by concatenating the columns of (7.16) six times. Due to the large system size, it is computationally very expensive to optimize for the code matrix using either the simulated annealing or cyclic column replacement methods. Therefore, we concatenate the columns of (7.16). This can be interpreted as a crowdsourcing system of 90 crowd workers consisting of 6 sub-systems with 15 workers each which are given the same task and their data is fused together. In the extreme case, if each of these sub-systems was of size one, it would correspond to a majority vote where all the workers are posed the same question. Fig. 7.4 shows the performance when $M = 8$ and N takes the two values: $N = 15$ and $N = 90$. These figures suggest that the gap in performance generally increases for larger system size. Similar observations hold for the beta model of crowds, see Figs. 7.5 and 7.6. Good codes perform better than majority vote as they diversify the binary questions which are asked to the workers. From extensive simulations, we found that the coding-based approach is not very sensitive to the choice of code matrix $\mathbf{A}$ as long as we have approximately equal number of ones and zeroes in every column. However, if we use any code randomly, performance may degrade substantially, especially when the quality of crowd is high. For example, consider a system consisting of $N = 15$ workers performing a $(M = 8)$-ary classification task. Their reliabilities are drawn from a spammer-hammer model and Fig. 7.7 shows the performance comparison between the coding-based approach using the optimal code matrix, majority-based approach and the coding-based approach using a random code matrix with equal number of ones and zeroes in every column. It can be observed that the performance of the coding-based approach with a random code matrix deteriorates for higher quality crowds.

Experimental ResultsFor Real Datasets

In this section, the proposed coding- based approach is tested on six publicly available Amazon Mechanical Turk data sets—quantized versions of the data sets in [130]: the anger, disgust, fear, joy, sadness and surprise datasets of the affective text task. Each of the data sets consist of 100 tasks with $N = 10$ workers taking part in each. Each worker reports a value between 0 and 100,

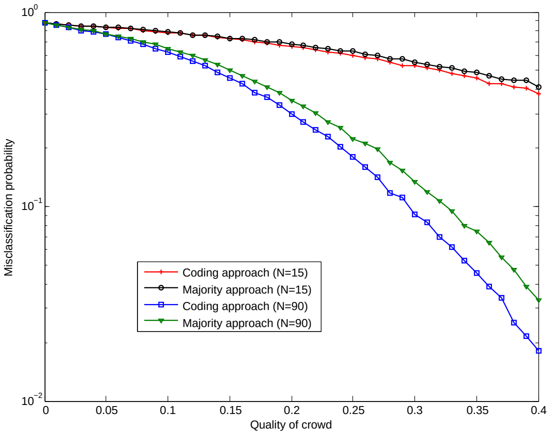


Fig. 7.4: Misclassification probability as a function of crowd quality using coding- and majority-based approaches with the spammer-hammer model, $(M = 8)$.

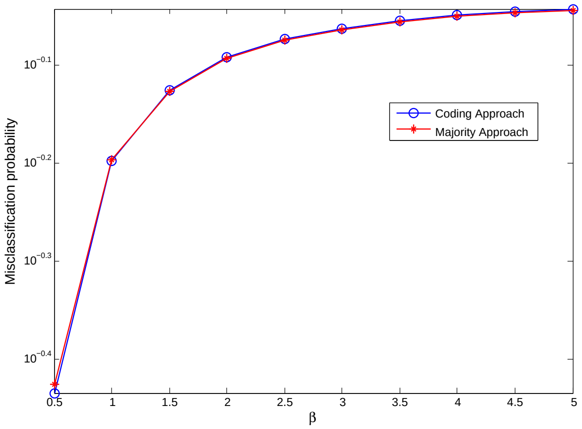


Fig. 7.5: Misclassification probability as a function of β using coding- and majority-based approaches with the Beta($\alpha = 0.5$, β) model, $(M = 4, N = 10)$.

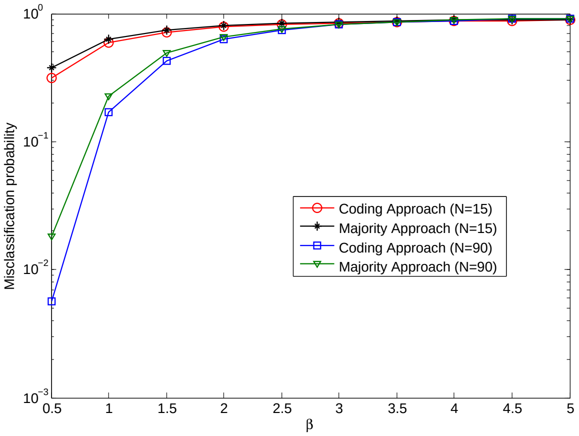


Fig. 7.6: Misclassification probability as a function of β using coding- and majority-based approaches with the Beta($\alpha = 0.5$, β) model, $(M = 8)$.

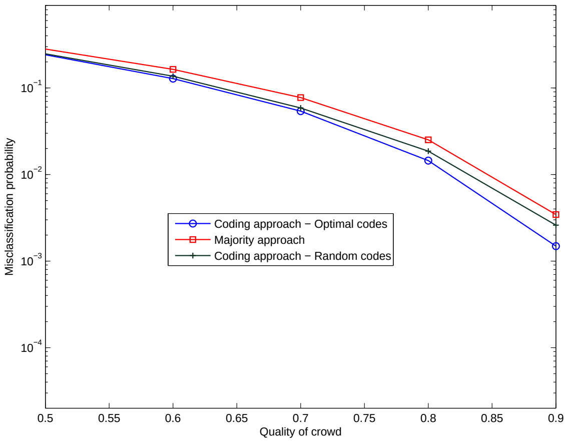


Fig. 7.7: Misclassification probability as a function of crowd quality using optimal code matrix, random code matrix for coding-based approach and majority approach with the spammer-hammer model, $(M = 8, N = 15)$.

Table 7.1: Fraction of errors using coding- and majority-based approaches

Dataset	Coding-based approach	Majority-based approach
Anger	0.31	0.31
Disgust	0.26	0.20
Fear	0.32	0.30
Joy	0.45	0.47
Sadness	0.37	0.39
Surprise	0.59	0.63

and there is a gold-standard value for each task. For the analysis, the values are quantized by dividing the range into $M = 8$ equal intervals. The majority -based approach is compared with the proposed coding-based approach. A good optimal code matrix for $N = 10$ and $M = 8$ is designed by simulated annealing [166]:

$$\mathbf{A} = [113, 139, 226, 77, 172, 74, 216, 30, 122]. \tag{7.17}$$

Table 7.1 compares the performance of the coding- and majority-based approaches. The values in Table 7.1 are the fraction of wrong decisions made, as compared with the gold-standard value. As indicated, the coding-based approach performs at least as well as the majority-based approach in 4 of 6 cases considered. The gap in performance is expected to increase as problem size M and crowd size N increase. Also, while it is true that the coding-based approach is only slightly better than the majority approach in the cases considered in Table 7.1, this comparison only shows the benefit of the proposed coding-based approach in terms of the fusion scheme. The datasets contain data for tasks where the workers have reported continuous values and, therefore, it does not capture the benefit of asking binary questions. This aspect is a major benefit of the proposed coding-based approach whose empirical testing is yet to be carried out.

7.4 Crowdsourcing System with Peer-dependent Reward Scheme

In this section, the crowdsourcing system is considered wherein the crowd workers are paired into groups of two and their reward value is based on the comparative performance among the paired workers [61]. This has been proposed as a method for increasing worker motivation.

7.4.1 Model

Two kinds of worker pairings are considered: competitive and teamwork. Both kinds can be captured by considering crowd reliability models where worker reliabilities are no longer independent as in Sec. 7.3. For simplicity, assume that N is even so that each worker j has a corresponding partner j_p and there are a total of $N/2$ pairs. The reliabilities of these paired workers are correlated with covariance ρ, which is assumed to be the same for all pairs due to the identical nature of workers. Also, workers within a pair are independent of the workers outside their pair. Hence:

$$cov(p_i, p_j) = \begin{cases} 0, & \text{if } i \neq j_p \\ \rho, & \text{if } i = j_p. \end{cases} \tag{7.18}$$

The value of ρ depends on whether workers are paired for teamwork or for competition [61]. An (N, M, μ, ρ) crowdsourcing system has N workers performing an M-ary classification task and having reliabilities p_j which are identical random variables with mean μ and covariance structure defined by (7.18).

7.4.2 Performance Characterization

In this section, the misclassification probability is derived when the crowd workers are correlated. As described above, this scenario takes place when the workers are paired with each other.

Proposition 7.4.1. *Consider an (N, M, μ, ρ) crowdsourcing system. The expected misclassification probability using code matrix $\mathbf{A}$ is:*

$$
\begin{aligned}
P_e(\mu,\rho) = \frac{1}{M}\sum_{\boldsymbol{i},l} C_{\boldsymbol{i}}^{l} \prod_{j=1}^{\frac{N}{2}} \Bigg[&(1-i_j)(1-i_{j_p}) + (1-i_j)(2i_{j_p}-1) \\
&\left(\mu a_{lj_p} + \frac{(1-\mu)}{(M-1)}\sum_{k\neq l} a_{kj_p}\right) + (1-i_{j_p})(2i_j-1) \\
&\left(\mu a_{lj} + \frac{(1-\mu)}{(M-1)}\sum_{k\neq l} a_{kj}\right) + (2i_j-1)(2i_{j_p}-1) \\
\Bigg((\rho+\mu^2)a_{lj}a_{lj_p} + \frac{\mu-(\rho+\mu^2)}{(M-1)} &\left(a_{lj}\sum_{k\neq l} a_{kj_p} + a_{lj_p}\sum_{k\neq l} a_{kj}\right) \\
&+ \frac{4r}{M^2}\sum_{k\neq l} a_{kj_p}\sum_{k\neq l} a_{kj}\Bigg)\Bigg]
\end{aligned}
\tag{7.19}
$$

where $r = \left(\frac{M}{2(M-1)}\right)^2 [(1-\mu)^2+\rho]$, $\boldsymbol{i} = [i_1, \cdots, i_N] \in \{0,1\}^N$ *is the received codeword and* $C_{\boldsymbol{i}}^{l}$ *is the cost associated with a global decision* H_l *when the received vector is* $\boldsymbol{i}$*. This cost is given in* (7.3).

Proof. Let $P_{e,\mathbf{p}}$ denote the misclassification probability given the reliabilities of the N workers. Then, if u_j denotes the bit sent by the worker j and the global decision is made using the Hamming distance criterion, $P_{e,\mathbf{p}} = \frac{1}{M}\sum_{\mathbf{i},l} P(\mathbf{u} = \mathbf{i}|H_l)C_{\mathbf{i}}^{l}$. Since local decisions are conditionally independent, $P(\mathbf{u} = \mathbf{i}|H_l) = \prod_{j=1}^{N} P(u_j = i_j|H_l)$. Further,

$$
\begin{aligned}
&P(u_j = i_j|H_l) = i_j P(u_j = 1|H_l) + (1-i_j)P(u_j = 0|H_l) \\
&= (1-i_j) + (2i_j-1)P(u_j = 1|H_l) \\
&= (1-i_j) + (2i_j-1)\sum_{k=1}^{M} a_{kj}P(y_j = k|H_l) \\
&= (1-i_j) + (2i_j-1)\left(p_j a_{lj} + \frac{(1-p_j)}{(M-1)}\sum_{k\neq l} a_{kj}\right)
\end{aligned}
$$

where y_j is the local decision made by worker j. Note that the reliabilities p_j of workers across pairs are independent while the workers within the pair are correlated according to (7.18). Therefore:

$$E\left[\prod_{j=1}^{N} P(u_j = i_j|H_l)\right] = \prod_{j=1}^{N/2} E\left[P(u_j = i_j|H_l)P(u_{j_p} = i_{j_p}|H_l)\right]$$
$$= \prod_{j=1}^{N/2} E\Bigg[\left((1-i_j) + \left(p_j a_{lj} + \tfrac{(1-p_j)}{(M-1)}\sum_{k\neq l} a_{kj}\right)(2i_j - 1)\right)$$
$$\left((1-i_{j_p}) + \left(p_{j_p} a_{lj_p} + \tfrac{(1-p_{j_p})}{(M-1)}\sum_{k\neq l} a_{kj_p}\right)(2i_{j_p} - 1)\right)\Bigg] \tag{7.20}$$

The above equation, correlation structure (7.18) and definition $r = \left(\frac{M}{2(M-1)}\right)^2[(1-\mu)^2+\rho]$ yield the desired result. $\square$

7.4.3 Majority Voting

As mentioned before, for the sake of comparison, error performance expressions are derived for the majority-based approach too. Consider majority vote, with N divisible by $2\log_2 M$.

Proposition 7.4.2. *Consider an (N, M, μ, ρ) crowdsourcing system. The expected misclassification probability using majority rule is:*

$$P_e(\mu,\rho) = 1 - \frac{1}{M}\left[1 + \sum_{j=\lfloor\frac{\tilde{N}}{2}+1\rfloor}^{\tilde{N}} b_j(\tilde{N},q,r)\left[(1-2q+r)^{(j-\frac{\tilde{N}}{2})} - r^{(j-\frac{\tilde{N}}{2})}\right]\right]^{\log_2 M},$$

where

$$b_j(\tilde{N},q,r) = \sum_{g=0}^{\lfloor\frac{\tilde{N}-j}{2}\rfloor}\binom{\frac{\tilde{N}}{2}}{g}\binom{\frac{\tilde{N}}{2}-g}{j+g-\frac{\tilde{N}}{2}}\left[2(q-r)\right]^{(\tilde{N}-j-2g)}(r - 2qr + r^2)^g,$$

$\tilde{N} = \frac{N}{\log_2 M}$, $q = \frac{M(1-\mu)}{2(M-1)}$, *and* $r = \left(\frac{M}{2(M-1)}\right)^2[(1-\mu)^2+\rho]$.

Proof. In a majority-based approach, $\tilde{N} = \frac{N}{\log_2 M}$ workers send information regarding the ith bit of their local decision, $i = 1, \dots, \log_2 M$. For a correct global decision, all bits have to be correct. Consider the ith bit and let $P^i_{c,\mathbf{p}}$ be the probability of the ith bit being correct given the reliabilities of the $\tilde{N}$ workers sending this bit. Also, assume that the paired workers send the same bit information. Then,

$$P^i_{c,\mathbf{p}} = \frac{P_d + 1 - P_f}{2}, \tag{7.21}$$

where P_d is the probability of detecting the ith bit as '1' when the true bit is '1' and P_f is the probability of detecting the ith bit as '1' when the true bit is '0'. Note that '0' and '1' are equiprobable since all the hypotheses are equiprobable. Under majority rule for this ith bit,

$$P_d = \sum_{j=\lfloor \frac{\tilde{N}}{2}+1 \rfloor}^{\tilde{N}} \sum_{\forall G_j} \prod_{k \in G_j} \left(1 - \frac{M(1-p_k)}{2(M-1)}\right) \prod_{k \notin G_j} \frac{M(1-p_k)}{2(M-1)}, \tag{7.22}$$

where G_j is a set of j out of $\tilde{N}$ workers who send bit value '1' and $\frac{M(1-p_k)}{2(M-1)}$ is the probability of the kth worker making a wrong decision for the ith bit. Similarly,

$$P_f = \sum_{j=\lfloor \frac{\tilde{N}}{2}+1 \rfloor}^{\tilde{N}} \sum_{\forall G_j} \prod_{k \in G_j} \frac{M(1-p_k)}{2(M-1)} \prod_{k \notin G_j} \left(1 - \frac{M(1-p_k)}{2(M-1)}\right), \tag{7.23}$$

Now the overall probability of correct decision is given by $P_{c,\mathbf{p}} = \prod_{i=1}^{\log_2 M} P^i_{c,\mathbf{p}}$. Since the workers across the groups are independent, the expected probability of correct decision P_c is:

$$P_c = \prod_{i=1}^{\log_2 M} E[P^i_{c,\mathbf{p}}], \tag{7.24}$$

where expectation is with respect to **p**. Within the same group, paired workers exist who are correlated and, therefore, the reliabilities are not independent. Let g pairs of workers be present in the set G^c_j, where $0 \leq g \leq \frac{\tilde{N}-j}{2}$. This implies that $2g$ workers in G^c_j are correlated with their partners in the same group and the remaining $(\tilde{N} - j - 2g)$ are correlated with their paired workers in G_j. Therefore, there are $(j + g - \frac{\tilde{N}}{2})$ pairs of correlated workers in G_j. The number of such divisions of workers into G_j and G_j^c such that there are exactly g pairs of correlated workers in G_j^c is determined as the number of ways of choosing g possible pairs from a total of $\frac{\tilde{N}}{2}$ pairs for G_j^c and $(j + g - \frac{\tilde{N}}{2})$ pairs of correlated workers from the remaining $(\frac{\tilde{N}}{2} - g)$ pairs for G_j. The remaining $(\tilde{N} - j - 2g)$ workers of $G_j^c(G_j)$ have their paired workers in $G_j(G_j^c)$ and this gives the following number of ways of such divisions:

$$N_g = \binom{\frac{\tilde{N}}{2}}{g}\binom{\frac{\tilde{N}}{2} - g}{j + g - \frac{\tilde{N}}{2}} 2^{(\tilde{N}-j-2g)}. \tag{7.25}$$

Define $q = \frac{M(1-\mu)}{2(M-1)}$ as the expected probability of a worker deciding a wrong bit and

$$\begin{aligned} r &= E\left[\frac{M(1-p_k)}{2(M-1)}\frac{M(1-p_{k_p})}{2(M-1)}\right] &(7.26)\\ &= \left(\frac{M}{2(M-1)}\right)^2 \left[(1-\mu)^2 + \rho\right] &(7.27)\end{aligned}$$

as the expected probability that both the workers in a correlated worker pair send wrong bit information. Taking the expectation of P_d and P_f (cf. (7.22), (7.23)) and using (7.25), we get:

$$\begin{aligned} E[P_d] &= \sum_{j=\lfloor\frac{\tilde{N}}{2}+1\rfloor}^{\tilde{N}} \sum_{g=0}^{\lfloor\frac{\tilde{N}-j}{2}\rfloor} \binom{\frac{\tilde{N}}{2}}{g}\binom{\frac{\tilde{N}}{2} - g}{j + g - \frac{\tilde{N}}{2}} \\ &\quad \left[2\left(q-r\right)\right]^{(\tilde{N}-j-2g)} r^g (1-2q+r)^{(j+g-\frac{\tilde{N}}{2})} \\ &= \sum_{j=\lfloor\frac{\tilde{N}}{2}+1\rfloor}^{\tilde{N}} b_j(\tilde{N}, q, r)(1-2q+r)^{(j-\frac{\tilde{N}}{2})} &(7.28)\end{aligned}$$

and

$$\begin{aligned} E[P_f] &= \sum_{j=\lfloor \frac{\tilde{N}}{2}+1\rfloor}^{\tilde{N}} \sum_{g=0}^{\lfloor \frac{\tilde{N}-j}{2}\rfloor} \binom{\frac{\tilde{N}}{2}}{g}\binom{\frac{\tilde{N}}{2}-g}{j+g-\frac{\tilde{N}}{2}} \\ &\quad [2\,(q-r)]^{(\tilde{N}-j-2g)}\, r^{(j+g-\frac{\tilde{N}}{2})}(1-2q+r)^{g} \\ &= \sum_{j=\lfloor \frac{\tilde{N}}{2}+1\rfloor}^{\tilde{N}} b_j(\tilde{N},q,r) r^{(j-\frac{\tilde{N}}{2})}. \end{aligned} \tag{7.29}$$

Using (7.21), (7.24), (7.28), and (7.29), gives the desired result. □

Some observations can be made here. First, when workers are not paired and instead perform the task individually, they are independent and $\rho = 0$. Therefore, $r = q^2$ and $P_e(\mu, 0)$ for majority- and coding-based approaches from Props. 7.4.1 and 7.4.2 reduces to the results from Sec. 7.3. Second, a crowd with mean $\mu = 1/M$, i.e., a crowd with workers of poor reliabilities on an average, has $q = 1/2$ and performs no better than random classification which gives $P_c(1/M, \rho) = 1/M$ for the majority case or $P_e(1/M, \rho) = \frac{(M-1)}{M}$. Similar observations can be made for the coding-based approach.

7.4.4 Performance Evaluation

Next, the performance of this system with peer-dependent reward scheme is evaluated. As mentioned before, such a scheme results in correlation among the crowd workers. Fig. 7.8 shows the performance of a system with peer-dependent reward scheme with varying correlation parameter (ρ_{corr}). Note that the plots are with respect to the correlation coefficient (ρ_{corr}) and the equations are with respect to the covariance parameter (ρ). The plots are for a system with N crowd workers and $M = 8$, i.e., performing 8-ary classification. As the figure suggests, the performance of the system improves when the workers are paired in the right manner. Note that when the crowdsourcing system with peer-dependent reward scheme has $\rho_{corr} = 0$, it reduces to the system with individual crowd workers considered in Sec. 7.3. Fig. 7.8 includes the system with independent

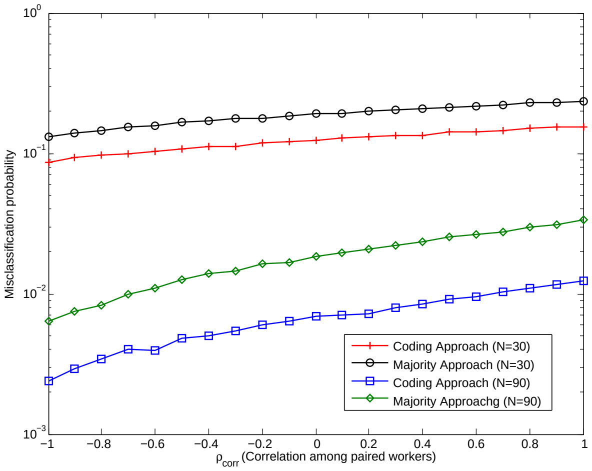


Fig. 7.8: Misclassification probability as a function of ρ_{corr} using coding- and majority-based approaches with the Beta($\alpha = 0.5$, $\beta = 0.5$) model, $(M = 8)$.

crowd workers as a special case when $\rho_{corr} = 0$. Also, these results suggest that having a negative correlation among workers results in better performance while a positive correlation deteriorates system performance. This observation can be attributed to the amount of diversity among the crowd workers. When $\rho_{corr} = 1$, the workers are correlated with each other while a correlation value of $\rho_{corr} = -1$ corresponds to a diverse set of workers which results in improved performance.

7.5 Crowdsourcing System with Common Sources of Information

In this section, the above model is further generalized by considering dependence among the observations of the crowd workers; previous sections had considered independent local observations by crowd workers. Although Sec. 7.4 considered paired workers with correlated reliabilities, here observations by crowd workers are dependent too. Crowd workers may have dependent observations when they share a common information source [111].

7.5.1 Model

One way to capture dependence among workers' observations is to assume that there is a set of latent groups $\{S_1, S_2, \cdots\}$, and each crowd worker j is assigned to one group S_{s_j} where $s_j \in \{1, 2, \cdots\}$ is a random variable indicating membership. Each group S_l is modeled to have an associated group reliability $r_l \in [0, 1]$ which determines the general reliability of the group. When a group represents the set of crowd workers who share a common source of information, the reliability r_l of the group S_l represents the reliability of the information source. Associated with every crowd worker j, then, is the worker's reliability $p_j \in [0, 1]$ which determines the worker's reliable use of the information.

We now describe a generative process for this model, the Multi-Source Sensing Model described in [111].

1. Draw $\lambda \sim$ GEM(κ), i.e., stick breaking construction with concentration κ. For this, the stick-breaking construction GEM(κ) (named after Griffiths, Engen and McCloskey) discussed in [111] is used. Specifically, in GEM(κ), a set of random variables $\gamma = \{\gamma_1, \gamma_2, \cdots\}$ are independently drawn from the beta distribution $\gamma_i \sim \beta(1, \kappa)$. They define the mixing weights λ of the group membership component such that $P(s_j = l|\gamma) = \lambda_l = \gamma_l \prod_{g=0}^{l-1}(1 - \gamma_g)$.

2. For each worker j, draw its group assignment $s_j|\lambda \sim$ Discrete(λ), where $s_j|\lambda \sim$ Discrete(λ) denotes a discrete distribution, which generates the value $s_j = l$ with probability λ_l.

3. For each group S_l, draw its group reliability $r_l \sim F(\mu)$, where $F(\mu)$ is the group reliability distribution with mean μ. Some possible examples are the beta model where reliabilities are beta distributed or the spammer-hammer model.

4. For each crowd worker j, draw their reliability $p_j \sim \beta\left(\frac{r_{s_j}}{1-r_{s_j}}, 1\right)$ such that $E[p_j] = r_{s_j}$

5. For each crowd worker j, draw his observation y_j based on reliability p_j and true hypothesis H_l (cf. (7.1))

Learning does not require prior knowledge on the number of groups, due to the stick-breaking process, but note the dependence among observations depends on the number of latent groups which itself is a function of the concentration parameter κ. Due to grouping of workers, reliabilities may not be independent random variables as in Sec. 7.4. An (N, M, μ, κ) crowdsourcing system consists of N grouped, but unpaired crowd workers performing M-ary classification, whereas an $(N, M, \mu, \rho, \kappa)$ crowdsourcing system has N grouped and paired crowd workers with pairing covariance ρ performing M-ary classification.

7.5.2 Performance Characterization

First consider crowd workers grouped by latent variables, but without peer-dependent rewards.

Proposition 7.5.1. *Consider an (N, M, μ, κ) crowdsourcing system. The expected misclassification probability using code matrix $\boldsymbol{A}$ is:*

$$P_e(\mu, \kappa) = \frac{1}{M} \sum_{\boldsymbol{i}, l, \boldsymbol{s}} C_{\boldsymbol{i}}^l P(\boldsymbol{S} = \boldsymbol{s} | H_l) \prod_{j=1}^{N} \left[\left(\mu a_{lj} + \frac{(1-\mu)}{(M-1)} \sum_{k \neq l} a_{kj} \right) (2i_j - 1) + (1 - i_j) \right],$$

where $\boldsymbol{i} = [i_1, \cdots, i_N] \in \{0, 1\}^N$ is the received codeword, $\boldsymbol{s} = [s_1, \cdots, s_N] \in \{0, \cdots, L-1\}^N$ is the group assignment, L is the number of groups and $C_{\boldsymbol{i}}^l$ is the cost associated with a global decision H_l when the received vector is $\mathbf{i}$. This cost is given in (7.3). *The probability $P(\boldsymbol{S} = \boldsymbol{s} | H_l)$ is a function of the concentration parameter κ and is:*

$$P(\boldsymbol{S} = \boldsymbol{s} | H_l) = \frac{1}{[\beta(1, \kappa)]^L} \prod_{l=0}^{L-1} \beta \left(n_l + 1, N + \kappa - \sum_{g=0}^{l} n_g \right)$$

where $\beta(\cdot, \cdot)$ is the beta function and n_l is the number of workers in the group S_l.

Proof. Let $P_{e,\mathbf{p}}$ denote the misclassification probability given the reliabilities of the N workers. Then, if u_j denotes the bit sent by the worker j and the global decision is made using the Hamming distance criterion, $P_{e,\mathbf{p}} = \frac{1}{M} \sum_{\mathbf{i}, l} P(\mathbf{u} = \mathbf{i} | H_l) C_{\mathbf{i}}^l$. Although local decisions are dependent, they are conditionally independent given the group assignments.

$$P(\mathbf{u}=\mathbf{i}|H_l) =$$

$$\sum_{\mathbf{s}} P(\mathbf{u}=\mathbf{i}|\mathbf{S}=\mathbf{s},H_l)P(\mathbf{S}=\mathbf{s}|H_l) = \sum_{\mathbf{s}} \left[\prod_{j=1}^{N} P(u_j=i_j|S_j=s_j,H_l)\right] P(\mathbf{S}=\mathbf{s}|H_l).$$

Further,

$$\begin{aligned}
& P(u_j=i_j|S_j=s_j,H_l) \\
= \;& i_j P(u_j=1|S_j=s_j,H_l) + (1-i_j)P(u_j=0|S_j=s_j,H_l) \\
= \;& (1-i_j) + (2i_j-1)P(u_j=1|S_j=s_j,H_l) \\
= \;& (1-i_j) + (2i_j-1)\sum_{k=1}^{M} a_{kj} P(y_j=k|S_j=s_j,H_l) \\
= \;& (1-i_j) + (2i_j-1)\left(p_j a_{lj} + \frac{(1-p_j)}{(M-1)}\sum_{k\neq l} a_{kj}\right)
\end{aligned}$$

and

$$P(\mathbf{S}=\mathbf{s}|H_l) = E_\lambda\left\{P(\mathbf{S}=\mathbf{s}|H_l,\lambda)\right\} \tag{7.30}$$

$$= E_\lambda\left\{\prod_{j=1}^{N} P(S_j=s_j|\lambda)\right\} = E_\lambda\left\{\prod_{j=1}^{N}\lambda_{s_j}\right\} \tag{7.31}$$

$$= E_\lambda\left\{\prod_{l=0}^{L-1}(\lambda_l)^{n_l}\right\} \tag{7.32}$$

$$= E_\gamma\left\{\gamma_0^{n_0}\prod_{l=1}^{L-1}\left[\gamma_l\prod_{g=1}^{l-1}(1-\gamma_g)\right]^{n_l}\right\} \tag{7.33}$$

$$= E_\gamma\left\{\gamma_0^{n_0}\prod_{l=1}^{L-1}\left[\gamma_l^{n_l}\prod_{g=1}^{l-1}(1-\gamma_g)^{n_l}\right]\right\} \tag{7.34}$$

$$= E_\gamma\left\{\gamma_0^{n_0}\gamma_1^{n_1}(1-\gamma_0)^{n_1}\gamma_2^{n_2}(1-\gamma_1)^{n_2}(1-\gamma_0)^{n_2}\cdots\right\}$$

$$= E_\gamma\left\{\prod_{l=0}^{L-1}\gamma_l^{n_l}(1-\gamma_l)^{(N-\sum_{g=0}^{l} n_g)}\right\} \tag{7.35}$$

Since γ are independently drawn from the beta distribution $\gamma_l \sim \beta(1, \kappa)$,

$$P(\mathbf{S} = \mathbf{s}|H_l) = \prod_{l=0}^{L-1} E\left[\gamma_l^{n_l}(1-\gamma_l)^{(N-\sum_{g=0}^{l} n_g)}\right]$$

$$= \prod_{l=0}^{L-1} \frac{\beta\left(n_l + 1, N + \kappa - \sum_{g=0}^{l} n_g\right)}{\beta(1,\kappa)} \tag{7.36}$$

$$= \frac{1}{[\beta(1,\kappa)]^L} \prod_{l=0}^{L-1} \beta\left(n_l + 1, N + \kappa - \sum_{g=0}^{l} n_g\right) \tag{7.37}$$

□

For workers without peer-dependent reward, the reliabilities p_j are independent random variables with mean r_{s_j}. Using the fact $E[r_{s_j}] = \mu$ yields the desired result.

Next, the peer-dependent reward scheme is introduced.

Proposition 7.5.2. *Consider an $(N, M, \mu, \rho, \kappa)$ crowdsourcing system. The expected misclassification probability using code matrix $\mathbf{A}$ is:*

$$\begin{aligned} P_e(\mu,\rho,\kappa) = \frac{1}{M} \sum_{\boldsymbol{i},l,\boldsymbol{s}} C_{\boldsymbol{i}}^l P(\boldsymbol{S} = \boldsymbol{s}|H_l) \prod_{j=1}^{\frac{N}{2}} \Bigg[& (1-i_j)(1-i_{j_p}) \\ & + (1-i_j)(2i_{j_p}-1)\left(\mu a_{lj_p} + \frac{(1-\mu)}{(M-1)} \sum_{k \neq l} a_{kj_p}\right) \\ & + (1-i_{j_p})(2i_j-1)\left(\mu a_{lj} + \frac{(1-\mu)}{(M-1)} \sum_{k \neq l} a_{kj}\right) \\ & + (2i_j-1)(2i_{j_p}-1)\Bigg((\rho+\mu^2) a_{lj} a_{lj_p} + \frac{\mu-(\rho+\mu^2)}{(M-1)} \\ & \left(a_{lj} \sum_{k \neq l} a_{kj_p} + a_{lj_p} \sum_{k \neq l} a_{kj}\right) + \frac{4r}{M^2} \sum_{k \neq l} a_{kj_p} \sum_{k \neq l} a_{kj}\Bigg)\Bigg] \end{aligned} \tag{7.38}$$

where $r = \left(\frac{M}{2(M-1)}\right)^2 [(1-\mu)^2 + \rho]$, $\boldsymbol{i} = [i_1, \cdots, i_N] \in \{0,1\}^N$ is the received codeword, $\boldsymbol{s} = [s_1, \cdots, s_N] \in \{0, \cdots, L-1\}^N$ is the group assignment, L is the number of groups and $C_{\boldsymbol{i}}^l$ is the cost associated with a global decision H_l when the received vector is $\boldsymbol{i}$. This cost is given in

(7.3). *The probability $P(\boldsymbol{S}=\boldsymbol{s}|H_l)$ is a function of the concentration parameter κ and is:*

$$P(\boldsymbol{S}=\boldsymbol{s}|H_l)=\frac{1}{[\beta(1,\kappa)]^L}\prod_{l=0}^{L-1}\beta\left(n_l+1,N+\kappa-\sum_{g=0}^{l}n_g\right) \tag{7.39}$$

where n_l is the number of workers in the group S_l.

Proof. The proof proceeds similarly to the proof of Prop. 7.5.1. However, since the workers are paired, they are correlated according to (7.18). This gives us

$$\begin{aligned}
&E\left[\prod_{j=1}^{N}P(u_j=i_j|S_j=s_j,H_l)\right]\\
&=\prod_{j=1}^{N/2}E\left[P(u_j=i_j|S_i=s_j,H_l)P(u_{j_p}=i_{j_p}|S_{j_p}=s_{j_p},H_l)\right]\\
&=\prod_{j=1}^{N/2}E\Bigg[\left((1-i_j)+\left(p_ja_{lj}+\frac{(1-p_j)}{(M-1)}\sum_{k\neq l}a_{kj}\right)(2i_j-1)\right)\\
&\left((1-i_{j_p})+\left(p_{j_p}a_{lj_p}+\frac{(1-p_{j_p})}{(M-1)}\sum_{k\neq l}a_{kj_p}\right)(2i_{j_p}-1)\right)\Bigg]
\end{aligned}$$

Using the above equation, correlation structure (7.18), and the definition of $r=\left(\frac{M}{2(M-1)}\right)^2[(1-\mu)^2+\rho]$ we get the desired result. ☐

Expressions for the majority approach can be derived as a special case of the coding-based approach by substituting the appropriate code matrix.

7.5.3 Performance Evaluation

Having analyzed the system with dependent observations among crowd workers, the performance of the system is now evaluated and the effect of the concentration parameter κ is analyzed. Fig. 7.9 shows the performance of the system as a function of the concentration parameter (κ). Note that a high value of κ implies independent observations while $\kappa=0$ implies completely dependent observations (all workers have a single source of information). The plots in Fig. 7.9 are for a system

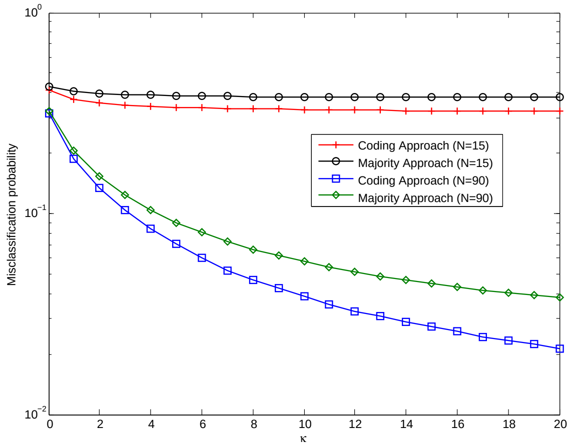


Fig. 7.9: Misclassification probability as a function of κ using coding- and majority-based approaches with the Beta($\alpha = 0.5$, $\beta = 0.5$) model, $(M = 8)$.

with uncorrelated workers (no peer-dependent reward scheme) performing an 8-ary classification $(M = 8)$.

As expected, one can observe that the system performance improves when the observations become independent. Also, the performance gap between majority- and coding-based approaches increases as the observations become independent. When the observations are dependent, all workers have similar observations on an average, so posing similar questions (majority-based approach) will perform similarly to posing diverse questions (coding-based approach). On the other hand, when the observations are independent, it is more informative to pose diverse questions to these workers as done in the coding-based approach. Therefore, we infer that *diversity is good* and the benefit of using coding increases when the observations are more diverse.

Similar observations can also be made for a system consisting of peer-dependent reward scheme with dependent observations. Fig. 7.10 shows the performance of such a system using both majority- and coding-based approaches. The plots are for a system with N crowd workers performing an 8-ary classification task $(M = 8)$. The group reliability distribution is assumed to be beta distribution and the correlation parameter is $\rho_{corr} = -0.5$.

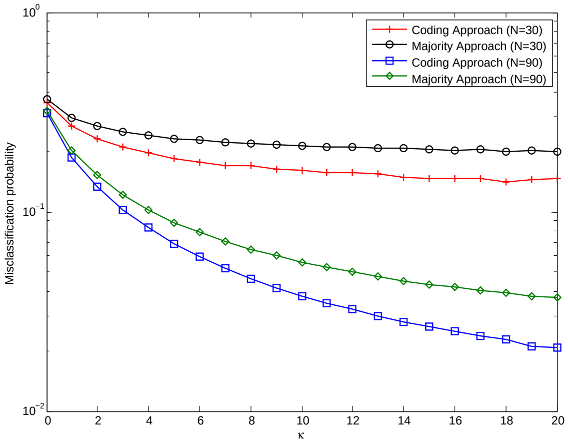


Fig. 7.10: Misclassification probability as a function of κ using coding- and majority-based approaches with the Beta($\alpha = 0.5$, $\beta = 0.5$) model and $\rho = -0.5$, $(M = 8)$.

7.6 Conclusion

In this chapter, the use of coding for reliable classification using unreliable crowd workers has been proposed. Using different crowdsourcing models, it has been shown that coding-based methods in crowdsourcing can more efficiently use human cognitive energy over traditional majority-based methods. Since minimum Hamming distance decoding is equivalent to MAP decoding in this setting, the anonymity of unreliable crowd workers is not a problem. We have shown that better crowds yield better performance. In other words, crowds which have higher reliabilities on an average perform better than the crowds with lower reliabilities. Although dependent observations have been considered in typical sensor networks, human decision makers in a crowdsourcing system give rise to multiple levels of dependencies. This work provides a mathematical framework to analyze the effect of such dependencies in crowdsourcing systems and provides some insights into the design aspects. By considering a model with peer-dependent reward scheme among crowd workers, it has been shown that pairing among workers can improve performance. Note that this aspect of rewards is a distinctive feature of crowdsourcing systems. It has also been shown that diversity in the system is desirable since the performance degrades when the worker observations are dependent.

The benefits of coding are especially large for applications where the number of classes is large, such as fine-grained image classification for building encyclopedias like Visipedia[2]. In such applications, one might need to classify among more than 161 breeds of dogs or 10000 species of birds. Designing easy-to-answer binary questions using the proposed scheme will greatly simplify the workers' tasks.

---

[2] http://www.vision.caltech.edu/visipedia/

CHAPTER 8

HUMAN-MACHINE INFERENCE NETWORKS (HUMAINS)

8.1 Introduction

In the previous chapters, systems with only sensors or only humans were studied. In this chapter, a framework for human-machine inference networks (HuMaINs) is presented. Note that the previous chapters focused on specific inference problems. An inference problem can also be referred to as problem-solving process, where the problem to be solved is that of inferring about a phenomenon. For example, an object classification task answers the question: “Which class does a given object belong to?". In other words, it solves the problem of classifying the given object into various possible classes.

In traditional economics, cognitive psychology, and artificial intelligence (AI) literature, the problem-solving process is described in terms of searching a problem space, which consists of various states of the problem, starting with the initial state and ending at the goal state [4]. Each path from the initial state represents a possible strategy which can be used. There could be multiple paths between the initial and the goal state which are the solutions to the problem. The focus here is on the case where there is a single path between the initial and the goal state. The problem-solving process is to identify this solution path among the multiple paths emanating from the initial state. Other paths lead to other goal states. Continuing with the example of object classification problem, the initial state is the unclassified object and the final state is the classified object. Each path from the initial state is a potential class for the given object, and the solution path is the true class. Identifying the solution path corresponds to the correct classification of the object. In this chapter, a problem-solving framework is considered and the benefits associated with human-machine collaboration to solve problems in an efficient manner are emphasized.

The first step for such a search is to determine the set of available strategies, i.e., the strategy space. For the object classification problem, this refers to identifying the set of possible classes that the object may belong to. The second step is to evaluate the strategies to determine the best strategy as the solution. For the object classification problem, this refers to observing the characteristics of the object and determining the true class by evaluating the possible classes. In traditional economic theory, a rational decision maker is assumed to have the knowledge of the set of possible alternatives[1], has the capability to evaluate the consequences of each alternative, and has a utility function which he/she tries to maximize to determine the optimal strategy [93]. However, it has long been debated that humans are not rational but are bounded rational agents. Under the bounded rationality framework [93, 127], decision makers are cognitively limited and have limited time, limited information, and limited resources. The set of alternatives is not completely known *a priori* nor are the decision makers perfectly aware of the consequences of choosing a particular alternative. Therefore, the decision maker might not always determine the best strategy for solving the problem.

On the other hand, machines are *rational* in the sense that they have stronger/larger memory for storing alternatives and have the computational capability to more accurately evaluate the consequences of a particular alternative. Therefore, a machine can aid a human in fast and accurate problem-solving. The goal of this chapter is to develop a framework for human-machine collaboration for problem-solving and illustrate its benefits. There are two basic ways in which a machine

---

[1] The terms strategy and alternative are used interchangeably.

can aid a human: in gaining knowledge about the set of alternatives, and in accurately evaluating the consequences of a chosen alternative. For example, in medical diagnosis, the human doctors might only look at a subset of possible diagnoses based on the symptoms, due to the cognitive limitations of humans. On the other hand, a machine with a database consisting of a much larger set of diseases can provide this human doctor with more exhaustive set of diagnoses by evaluating the symptoms in a timely manner. Such a machine can support the doctor in recognizing some of the rare diseases and also provide corresponding recommendations which could have been overlooked by the doctor. This is partly the idea behind IBM's Watson, M. D.[2] Another example is the task of pattern recognition. While humans are good at identifying new patterns, machines are good at searching for specific patterns. Therefore, in a pattern recognition task, humans can provide new patterns which the machines can search for.

As stated by Lubart in [88], when the machine supports the human in determining the set of alternatives, the machine is acting as a *coach* to the human to discover new alternatives. When the machine is helping the human by evaluating the effects of a particular alternative, the machine is acting as a *colleague* to the human in solving a problem. We discuss these two problems in this chapter, with more focus on the first one, to illustrate the benefits of using human machine collaboration for problem-solving.

8.2 Knowledge Discovery Problem

In this section, the first way in which a machine can help a human is explored, by providing him/her with new information, so that he/she can make a well-informed decision by evaluating the available alternatives. This *knowledge discovery problem* occurs when the human is undergoing little-d discovery or P-discovery. This terminology is inspired by the terminology used in the *creativity* literature, where little-c creativity means novelty with respect only to the mind of the individual concerned or everyday creativity which can be found in nearly all people, and big-C

---

[2] http://www/ibm.com/smarterplanet/us/en/ibmwatson/

Creativity means novelty with respect to the whole of previous history or eminent creativity, which is reserved for the great [21]. The knowledge discovery problem considered here is different from the human-machine collaboration for scientific discovery in the AI area [75, 82, 144] that can be referred to as big D-Discovery or H-discovery.

The output of the little-d discovery process considered here is expansion of knowledge base by adding additional elements into the knowledge base. While new elements are discovered and added into the knowledge base, it is important to understand the effect of discovering new elements. In creativity literature, generation of a product or service is judged based on its novelty and also its appropriateness, usefulness, or value to a knowledgeable social group [123]. While novelty of the element implies that it was previously unknown and therefore increases the size of the knowledge base, quality refers to the value it brings to the knowledge base when it is added. Therefore, size (number of distinct elements) and quality of the knowledge base are used as metrics in evaluating the discovery process. For a given set of elements in the knowledge base, size is defined as the total number of elements in the knowledge base and quality is defined as the total value of the elements in the knowledge base.

8.2.1 Problem Formulation

The knowledge discovery problem can be mathematically formulated as follows. Consider a set of M total elements, denoted by Θ, that are to be learnt by a human. This is the set that represents the entire knowledge base. The human is initially only aware of the presence of a subset $\Theta_0 \subset \Theta$ of these M possibilities. He/She learns the other elements over multiple iterations. At each iteration, the unknown element θ can be one of the M possible values according to a prior probability mass function (pmf) $\mathbf{p} = [p_1, \ldots, p_M]$. A machine makes a noisy observation of this element resulting in a noisy estimate of this element. This noisy estimate is provided to the human, who updates his/her element set by adding it to his/her current knowledge base. In other words, the machine observes a noisy version of element $\theta = m$ with probability p_m and provides its estimate based on this observation to the human. Note that the machine presents its outcome to the human without any information regarding the initial subset of elements known by the human. Let the underlying element θ_t be observed at iteration t. The machine makes an observation x_t^M corrupted by noise which is i.i.d. across time. Based on this observation, the machine infers the element to be $\hat{\theta}_t \in \Theta$ as follows:

$$\hat{\theta}_t = \arg\max_{\theta \in \Theta} p(\theta | x_t^M),$$

where $p(\theta | x_t^M)$ is the posterior pmf given the observation. The human's updated element set is $\Theta_t = \Theta_{t-1} \cup \hat{\theta}_t$. Each new element is characterized in terms of two basic performance measures: novelty and quality.

8.2.2 Size of Knowledge Base

As mentioned before, size is measured by the total number of distinct elements that are known to the human at any given time. This includes both the prior set of elements known to the human before the start of the discovery process as well as the elements learnt with the help of a machine during the discovery process. Note that the number of elements discovered by the human during the discovery process can be easily determined by subtracting the initial size of the knowledge base from the size of the current knowledge base. Let $N_t = |\Theta_t|$ denote the total number of elements in the human's knowledge base at iteration t. Then, the number of elements discovered by the human after t iterations is given by $D_t = |\Theta_t| - |\Theta_0|$. In this chapter, however, we focus only on the size that gives the total number of distinct elements known to the human.

First, we make some intuitive observations that are later verified using theoretical analysis and simulation results. The learning rate, defined as the rate at which knowledge set grows, depends on the pmf $\mathbf{p}$, the initial set Θ_0, and the noise distribution. The performance, characterized in terms of this learning rate, is best for uniform prior due to maximum uncertainty (entropy), when the initial knowledge set Θ_0 is arbitrary. However, for cases when Θ_0 contains rare elements (low pmf value), the knowledge discovery is expected to be fast as the unknown elements are of high probability and can be quickly learnt by the human. These intuitive observations can be observed in our analytical results. Also, for noisy observations, it might be possible that higher noise variance might result in faster total discovery as noisy machine's observations can cause false decisions at the machine helping the discovery of new elements (typically of low probability) by the human. While false decisions might cause negative consequences if one acts upon them, they might be helpful when the goal is just to learn all possible elements. In such cases, false decisions at the machine might help add new elements into the human's knowledge base.

Let the misclassification probabilities due to noise corrupted unreliable observations at the machine be

$$r_{mn} \triangleq Pr(\hat{\theta}_t = n|\theta_t = m)$$

for $m, n = 1, \ldots, M$, which are assumed to be the same for all t. Therefore, the posterior pmf $\tilde{\mathbf{p}} = [\tilde{p}_1, \ldots, \tilde{p}_M]$ of the elements is the new prior distribution of the elements observed by the human. This pmf is given as

$$\tilde{p}_m = \sum_{n=1}^{M} p_n r_{nm}.$$

An asymmetric *channel* $\mathbf{R} \triangleq \{r_{mn}\}$ between machine's input and output can represent the difficulty associated with the identification of elements by the machine. More explicitly, the machine can identify some elements more precisely than others, which would result in an asymmetric channel $\mathbf{R}$. Some elements can be more difficult to identify/discover than others [7–9].

Proposition 8.2.1. *For the machine-aided knowledge discovery problem with a pmf* $\mathbf{p}$ *and misclassification channel* $\mathbf{R}$ *at the machine, the expected size of knowledge base after* T *iterations is given by*

$$E\left[N_T|\Theta_0\right] = |\Theta_0| - \sum_{k=1}^{T} \sum_{\theta \notin \Theta_0} (-1)^k \binom{T}{k} \tilde{p}_\theta^k, \tag{8.1}$$

and the expected number of elements discovered by the human after T *iterations is given by*

$$E\left[D_T|\Theta_0\right] = \sum_{k=1}^{T} \sum_{\theta \notin \Theta_0} (-1)^{k+1} \binom{T}{k} \tilde{p}_\theta^k. \tag{8.2}$$

Proof. The number of elements are updated as follows

$$N_t = \begin{cases} N_{t-1} & \text{if } \hat{\theta}_t \in \Theta_{t-1} \\ N_{t-1}+1 & \text{if } \hat{\theta}_t \notin \Theta_{t-1}, \end{cases} \tag{8.3}$$

resulting in a Markov Chain:

$$N_t = \begin{cases} N_{t-1} & \text{with probability } \sum_{\theta\in\Theta_{t-1}} \tilde{p}_\theta \\ N_{t-1}+1 & \text{with probability } 1-\sum_{\theta\in\Theta_{t-1}} \tilde{p}_\theta. \end{cases} \tag{8.4}$$

Therefore,

$$\begin{aligned} E\left[N_t|\Theta_{t-1}\right] &= |\Theta_{t-1}| \sum_{\theta\in\Theta_{t-1}} \tilde{p}_\theta + (|\Theta_{t-1}|+1)(1-\tilde{p}_\theta) \\ &= |\Theta_{t-1}| + 1 - \sum_{\theta\in\Theta_{t-1}} \tilde{p}_\theta. \end{aligned} \tag{8.5}$$

For $t=1$,

$$E\left[N_1|\Theta_0\right] = m_0 + 1 - \sum_{\theta\in\Theta_0} \tilde{p}_\theta, \tag{8.6}$$

where $|\Theta_0| = m_0$. Next, for $t=2$,

$$E\left[N_2|\Theta_1\right] = N_1 + 1 - \sum_{\theta\in\Theta_1} \tilde{p}_\theta \tag{8.7}$$

$$= N_1 + 1 - \sum_{\theta\in\Theta_0} \tilde{p}_\theta - \sum_{\theta\in\Theta_1\backslash\Theta_0} \tilde{p}_\theta. \tag{8.8}$$

Recursively, we have for $t \geq 2$,

$$\begin{aligned}
& E\left[N_t | \Theta_{t-2}\right] \\
= \; & E\left[N_{t-1} + 1 - \sum_{\theta \in \Theta_{t-2}} \tilde{p}_\theta - \sum_{\theta \in \Theta_{t-1} \backslash \Theta_{t-2}} \tilde{p}_\theta \Big| \Theta_{t-2}\right] \\
= \; & |\Theta_{t-2}| + 2 \sum_{\theta \notin \Theta_{t-2}} \tilde{p}_\theta - E\left[\sum_{\theta \in \Theta_{t-1} \backslash \Theta_{t-2}} \tilde{p}_\theta \Big| \Theta_{t-2}\right].
\end{aligned} \tag{8.9}$$

An additional element is added into Θ_t only when it is not originally present in Θ_{t-1} (it is novel), which is with probability $\sum_{\theta \notin \Theta_{t-1}} \tilde{p}_\theta$. Under this condition that an element has been added, each $\theta \notin \Theta_{t-2}$ occurs with probability $\tilde{p}_\theta / \sum_{\theta \notin \Theta_{t-2}} \tilde{p}_\theta$, which reduces (8.9) to the following

$$E\left[N_t | \Theta_{t-2}\right] = |\Theta_{t-2}| + 2 \sum_{\theta \notin \Theta_{t-2}} \tilde{p}_\theta - \sum_{\theta \notin \Theta_{t-2}} \tilde{p}_\theta^2. \tag{8.10}$$

Continuing further, for a general T

$$E\left[N_T | \Theta_0\right] = |\Theta_0| - \sum_{k=1}^{T} \sum_{\theta \notin \Theta_0} (-1)^k \binom{T}{k} \tilde{p}_\theta^k. \tag{8.11}$$

$\square$

Some observations can be made from the above expressions:

- As seen from (8.1), if the rare elements are already known (θ_0 consists of elements whose prior probability is low), then the convergence of the discovery process is faster and all the elements can be learnt quicker by the human. This observation is intuitively expected as we had previously stated.
- For the special case of uniformly distributed prior pmf and perfect observations ($\mathbf{R}$ is identity

matrix),

$$E\left[N_T|\Theta_0\right] = |\Theta_0| - \sum_{k=1}^{T} \sum_{\theta \notin \Theta_0} (-1)^k \binom{T}{k} \frac{1}{M^k} \tag{8.12}$$

$$E\left[N_T|\Theta_0\right] = |\Theta_0| - \sum_{\theta \notin \Theta_0} \sum_{k=1}^{T} \binom{T}{k} \left(\frac{-1}{M}\right)^k \tag{8.13}$$

$$E\left[N_T|\Theta_0\right] = |\Theta_0| - \sum_{\theta \notin \Theta_0} \left(\left(1 - \frac{1}{M}\right)^T - 1\right) \tag{8.14}$$

$$E\left[N_T|\Theta_0\right] = |\Theta_0| - (M - |\Theta_0|)\left(1 - \frac{1}{M}\right)^T + (M - |\Theta_0|) \tag{8.15}$$

$$E\left[\rho_T|\rho_0\right] = 1 - (1 - \rho_0)\left(1 - \frac{1}{M}\right)^T, \tag{8.16}$$

where $\rho_t = N_t/M$ is the fraction of elements known by iteration t.

- Rate of knowledge discovery (rate of convergence of $E[\rho_t|\rho_0]$ to 1) increases monotonically in ρ_0.

- Growth rate of knowledge discovery is exponential in T. For the case of uniformly distributed prior pmf and perfect observations, this rate is given by

$$\lim_{T\to\infty} \frac{1}{T} \log(1 - E[\rho_T|\rho_0]) = -\log\left(1 - \frac{1}{M}\right). \tag{8.17}$$

  Therefore, the convergence is faster as M decreases. In other words, when the total number of elements is less, they can be learnt faster, as expected.

For simulation purposes, the following parameters are used in the experiment: $\Theta = \{1, \ldots, M\}$, $p_\theta(m) = \binom{M-1}{m-1} p^{m-1}(1-p)^{M-m}$ for $m = 1, \ldots, M$, and the noise channel is M-ary symmetric channel with crossover probability r. In other words, we are looking at discovering a total set of M elements. To characterize the prior using a single parameter, we use a binomial prior with parameters M and p. The symbols are ordered in decreasing order of their prior probability of being discovered.

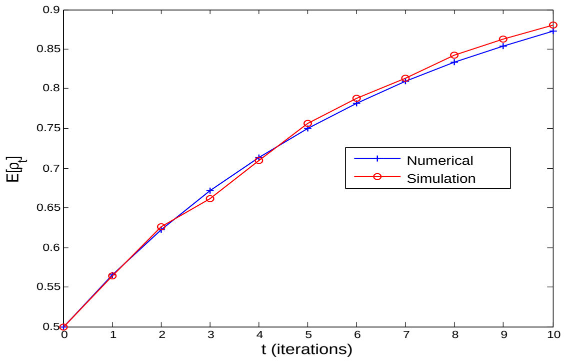


Fig. 8.1: Numerical and simulations results of fraction of element set (ρ_t) ($M = 4$, $\Theta_0 = [1, 2]$, $p = 0.2$, and $r = 0.1$).

Fig. 8.1 shows the numerical and simulation results that corroborate our analytical results. The simulation results are averaged over $N_{mc} = 500$ Monte-Carlo runs. From this figure, we can see that the theoretical expressions derived in this chapter capture the behavior of the knowledge discovery process. The fraction of elements in the knowledge base increase at an exponential rate. As can be observed from Fig. 8.2, the performance depends on the initial set Θ_0 and the performance (in terms of knowledge discovery rate) is higher (and also better than the uniform case) when the initial elements are the least probable ones ($\Theta_0 = [3, 4]$ in this example). This is intuitively true since, if the human already knows the rare elements, he/she can discover the others at a faster rate.

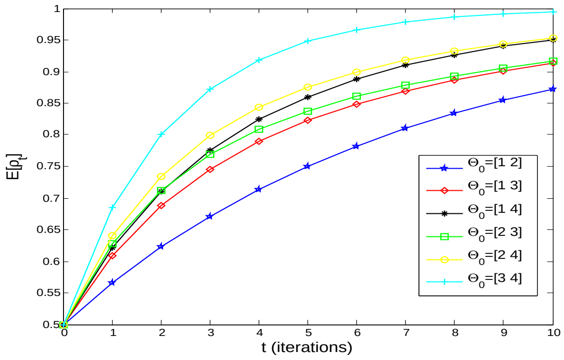


Fig. 8.2: Performance variation with different initial sets Θ_0 ($M = 4$, $p = 0.2$, and $r = 0.1$).

Fig. 8.3 shows the performance with varying noise value r where higher r implies noisier data at the machine. When the human is already aware of the most probable elements (refer to Fig. 8.4), noise has the positive effect of helping the human in discovering the lesser probable elements at a faster rate implying that ***a noisy machine can help us in discovering new elements!*** Such an observation is related in concept to the phenomenon of Stochastic Resonance (SR) or noise-enhanced signal processing [32], where addition of noise can improve the system performance of some non-linear suboptimal systems. It has been shown that the performance of some detection and estimation systems can be improved by adding noise [31]. The exact form of noise, characterized in terms of its pdf has been shown to be dependent on the problem of interest [31]. For example, for a detection problem in the Neyman-Pearson setting, it has been shown that a two-peak noise is optimal and for the Bayesian setting, one-peak noise has been proved to be optimal [31]. While the similarities are evident, further analysis is needed to mathematize this relation and understand the optimal noise that results in best performance. This thesis only shows the effect that noise can have on the discovery process by considering a symmetric noise model. Determining the type (pdf) of optimal noise will be considered in the future.

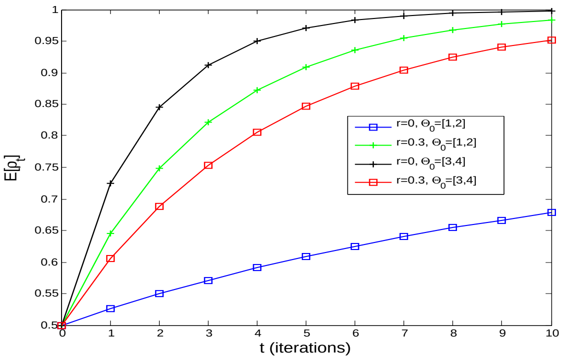


Fig. 8.3: Effect of noise ($M = 4$, and $p = 0.2$).

Fig. 8.5 shows how the performance varies with prior distribution. The effect of prior distribution on the performance is characterized by (8.1). By varying the parameter p, we can simulate different prior probability mass functions. As p tends to 0.5, the prior pmf moves towards a uniform prior. Therefore, we can observe that for high values of p (0.4), the discovery process is slowest. However, there is no general trend that can be observed as p varies from 0.1 to 0.4.

For the simple case of uniform prior and perfect observations at the machine, we evaluate the

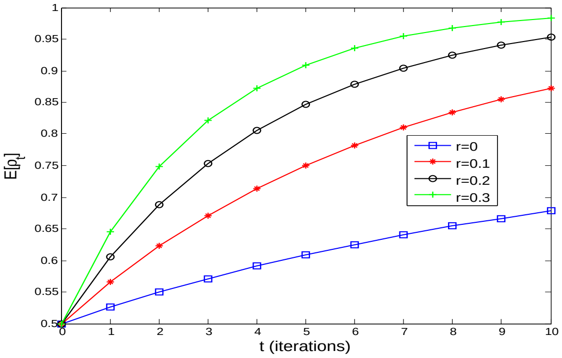


Fig. 8.4: Effect of noise ($M = 4$, $p = 0.2$, and $\Theta_0 = [1, 2]$).

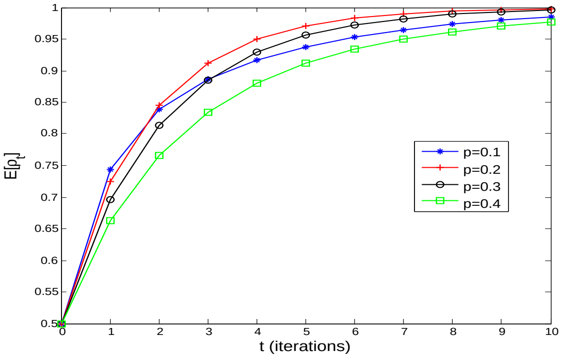


Fig. 8.5: Effect of different prior distribution ($M = 4$, $r = 0$, and $\Theta_0 = [3, 4]$).

performance with varying M and ρ_0 in Figs. 8.6 and Figs. 8.7, respectively. As can be observed from Fig. 8.6, the discovery process gets slower when the number of total elements (M) is higher. This corroborates the observation that the convergence is faster as M decreases. Fig. 8.7 shows the discovery process with different number of initially known elements. Clearly, one can learn all the elements faster, if he/she already knows a good portion of them.

8.2.3 Quality of Knowledge Base

In many cases, not only is the discovery of new elements important but also the quality of the discovered element. Consider the case where each element $\theta \in \Theta$ has a corresponding quality factor q_θ which determines the value of discovering θ. For example, while new restaurants can be

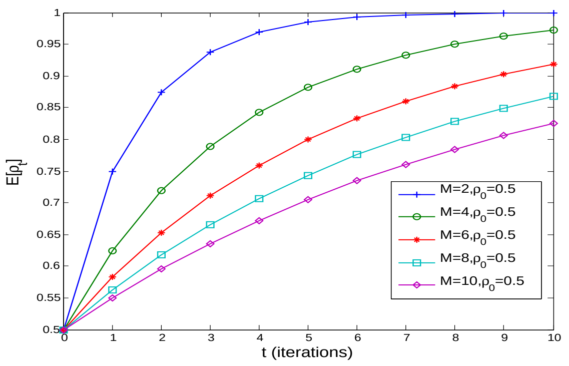


Fig. 8.6: Expected fraction of element set with iterations ($\rho_0 = 0.5$, varying M)

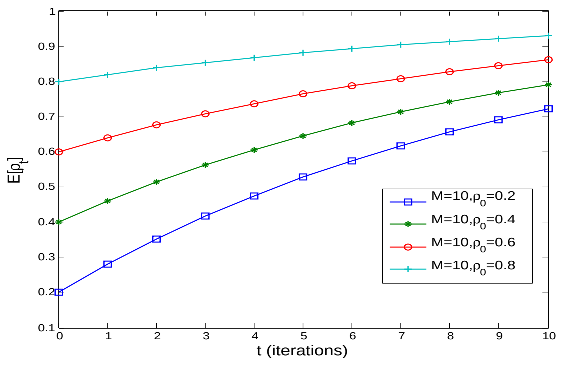


Fig. 8.7: Expected fraction of element set with iterations ($M = 10$, varying ρ_0)

discovered using Yelp[3], the reviews it has represent its value. While the discovery is novel, it need not be of good quality. Let $\mathbf{q} = [q_1, \ldots, q_M]$ be the quality vector and $Q_t = \sum_{\theta \in \Theta_t} q_\theta$ denote the quality of the elements discovered after time t.

Definition 8.2.2. *The k-th order quality-prevalence function $D^k_{\Theta_0}(\tilde{\mathbf{p}}, \mathbf{q})$ is defined as the inner product between $\mathbf{q}$ (quality vector) and element-wise kth power of $\tilde{\mathbf{p}}$ (probability vector) over the subset Θ_0^C ($\Theta \backslash \Theta_0$): $D^k_{\Theta_0}(\tilde{\mathbf{p}}, \mathbf{q}) := \sum_{\theta \notin \Theta_0} q_\theta \tilde{p}^k_\theta$.*

These functions evaluate the degree of alignment between the probability vector and the quality vector. In other words, these functions have high value when the high probable elements are also of high quality (aligned in the same *direction*) and low value when the low probable elements are of high quality (aligned in the opposite *direction*). These functions evaluate the inner product of

---

[3] http://www.yelp.com/

the quality vector with element-wise powers of probability vector, thereby bringing a geometric notion of alignment.

Proposition 8.2.3. *For the problem of machine-aided knowledge discovery problem, the expected quality of elements in the knowledge base after T iterations is given by*

$$E\left[Q_T|\Theta_0\right] = Q_0 - \sum_{k=1}^{T}(-1)^k \binom{T}{k} D^k_{\Theta_0}(\tilde{\mathbf{p}}, \mathbf{q}), \tag{8.18}$$

where $D^k_{\Theta_0}(\tilde{\mathbf{p}}, \mathbf{q})$ is the kth order quality-prevalence function.

Proof. Let $Q_t = \sum_{\theta \in \Theta_t} q_\theta$ denote the quality of the elements known after time t. Then we have the following relation:

$$Q_t = \begin{cases} Q_{t-1} & \text{if } \hat{\theta}_t \in \Theta_{t-1} \\ Q_{t-1} + q_{\hat{\theta}_t} & \text{if } \hat{\theta}_t \notin \Theta_{t-1}. \end{cases} \tag{8.19}$$

This implies

$$Q_t = \begin{cases} Q_{t-1} & \text{with probability } \sum_{\theta \in \Theta_{t-1}} \tilde{p}_\theta \\ Q_{t-1} + q_\theta & \text{with probability } \tilde{p}_\theta \text{, for every } \theta \notin \Theta_{t-1}. \end{cases} \tag{8.20}$$

Therefore,

$$\begin{aligned} E\left[Q_t|\Theta_{t-1}\right] &= Q_{t-1} \sum_{\theta \in \Theta_{t-1}} \tilde{p}_\theta + \sum_{\theta \notin \Theta_{t-1}} (Q_{t-1} + q_\theta)\tilde{p}_\theta & (8.21) \\ &= Q_{t-1} + \sum_{\forall \theta} q_\theta \tilde{p}_\theta - \sum_{\theta \in \Theta_{t-1}} q_\theta \tilde{p}_\theta. & (8.22) \end{aligned}$$

Following analysis similar to the analysis in proof of Prop. 8.2.1, we have for $t = 1$,

$$E\left[Q_1|\Theta_0\right] = Q_0 + \sum_{\theta} q_\theta \tilde{p}_\theta - \sum_{\theta \in \Theta_0} q_\theta \tilde{p}_\theta, \tag{8.23}$$

where $Q_0 = \sum_{\theta \in \Theta_0} q_\theta$ is the quality of the initial elements. Similarly for $t = 2$,

$$\begin{aligned} E\left[Q_2 | \Theta_1\right] &= Q_1 + \sum_{\theta} q_\theta \tilde{p}_\theta - \sum_{\theta \in \Theta_1} q_\theta \tilde{p}_\theta && (8.24) \\ &= Q_1 + \sum_{\theta} q_\theta \tilde{p}_\theta - \sum_{\theta \in \Theta_0} q_\theta \tilde{p}_\theta - \sum_{\theta \in \Theta_1 \setminus \Theta_0} q_\theta \tilde{p}_\theta . && (8.25) \end{aligned}$$

Therefore, we have

$$\begin{aligned} E\left[Q_2 | \Theta_0\right] &= E\left[Q_1 + \sum_{\theta} q_\theta \tilde{p}_\theta - \sum_{\theta \in \Theta_0} q_\theta \tilde{p}_\theta - \sum_{\theta \in \Theta_1 \setminus \Theta_0} q_\theta \tilde{p}_\theta \middle| \Theta_0 \right] && (8.26) \\ &= Q_0 + 2 \sum_{\theta \notin \Theta_0} q_\theta \tilde{p}_\theta - E\left[\sum_{\theta \in \Theta_1 \setminus \Theta_0} q_\theta \tilde{p}_\theta \middle| \Theta_0 \right] && (8.27) \\ &= Q_0 + 2 \sum_{\theta \notin \Theta_0} q_\theta \tilde{p}_\theta - \sum_{\theta \notin \Theta_0} q_\theta \tilde{p}_\theta^2 . && (8.28) \end{aligned}$$

Continuing further, we have, for a general T

$$E\left[Q_T | \Theta_0\right] = Q_0 - \sum_{k=1}^{T} \sum_{\theta \notin \Theta_0} (-1)^k \binom{T}{k} q_\theta \tilde{p}_\theta^k . \qquad (8.29)$$

□

The two extreme possibilities are when $\mathbf{q}$ is aligned either in the same direction as probability vector $\mathbf{p}$ or in the opposite direction. For the previous example, Fig. 8.8 shows the quality of the discovered elements for these two extreme cases which confirms our understanding.

8.2.4 Empirical Observations

Eurekometrics [8] is the study of nature of discovery. It has been shown using empirical results that discovery of scientific output increases exponentially, or more properly, a logistic growth curve. Arbesman and his colleagues [7–9] have empirically quantified the discovery process. By considering three different scientific disciplines: mammalian species, chemical elements, and mi-

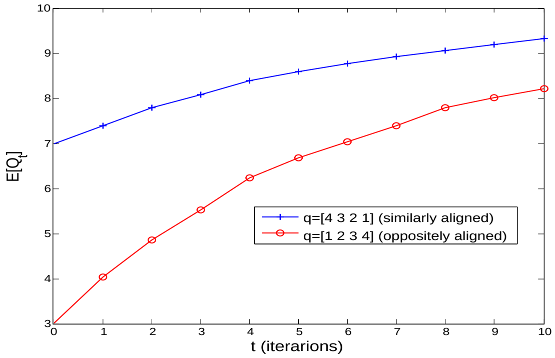


Fig. 8.8: Expected quality with time ($M = 4$, $r = 0.1$, $p = 0.2$, and $\Theta_0 = [1, 2]$).

nor planets, Arbesman et al. show that ease of scientific discovery is exponential resulting in an approximate logistic curve for the number of discoveries with time [7]. In the theoretical investigation performed here, the ease of discovery corresponds to the pmf $\mathbf{p}$ that denotes the difficulty associated with discovering an element. When all elements are assumed to be of equal difficulty, empirical results suggest a logistic curve for the number of discovered elements D_T [7]:

$$D_T \approx \frac{K}{1 + Ae^{-r_0 T}} \tag{8.30}$$

where K is the limiting size or the maximum number of elements that can be discovered, A is the fitting constant, and r_0 is the growth rate of scientific output. For a small value of A, this can be approximated as

$$D_T \approx K(1 - Ae^{-r_0 T}) \tag{8.31}$$

$$\Longleftrightarrow \rho_T^{emp} \approx 1 - Ae^{-r_0 T} \tag{8.32}$$

where $\rho_T^{emp} = D_T/K$ is the fraction of discovered elements. Observe that (8.32) matches the expression (8.16) derived for the mathematical model of discovery process developed here. This suggests that the mathematical model developed herein is in coherence with the empirical observations. In the future, this relation will be further explored to understand and interpret the parameters.

8.3 Solution Search Problem

In this section, another example of human-machine collaboration to solve difficult problems is introduced. As mentioned in Sec. 8.1, any problem-solving mechanism can be interpreted as a search for the 'best' alternative among multiple alternatives. This can be represented as search for the maximum of a function $f(x)$ among all alternatives $x \in \mathcal{X}$. Here, the function $f(x)$ quantifies the consequences of choosing an alternative x and, therefore, does not have an explicit known form. For example, while playing a complex game such as chess, one is trying to maximize the probability of winning which depends on the strategy chosen. Since humans have limited cognitive capabilities, they might not be well-equipped to perform a quick and accurate evaluation of the function for a given alternative. Therefore, a machine can aid the human by providing an accurate characterization of the consequences of choosing a particular alternative. This human-machine partnership falls under the *machine as a colleague* category for problem-solving. Such colleague-based partnership is the basis for the *Advanced Chess Tournaments* (also referred to as Centaur or Cyborg Chess) [142].

8.3.1 Mathematical Formulation

Consider a problem represented as $f(\mathbf{x})$ where each alternative $\mathbf{x}$ is a possible strategy for solving the problem. Solving the problem requires N elements (of a decision vector), i.e., each alternative contains N variables $\mathbf{x} = [x_1, \ldots, x_N]$. Each of these variables x_i has to be chosen from a set $\mathcal{X}_i$. The goal is to determine $\mathbf{x}^0$ which results in the maximum value of $f(\mathbf{x}^0) = f^0$. Since each strategy $\mathbf{x}$ is of dimension N (typically high), it is difficult for a human to evaluate the consequences of choosing a particular strategy. Therefore, he/she takes the help of a machine to accurately evaluate the function value. The human has a preference order $p(\mathbf{x})$ on the choice of possible solutions to the problem. This preference could be a result of experience in dealing with similar problems or simply due to his/her intuition. For example, in chess, this preference order depends on how good the player is.

Definition 8.3.1. *A problem is a Lipschitz continuous problem if the "distance" between the consequences (in the consequences space) of two different strategies is bounded by a constant times the "distance" between the strategies (in the strategy space).*

As can be seen from the above definition, most of the problems that we face in real life are Lipschitz continuous since changing our strategy by a small amount usually changes the consequences in a limited manner. For example, choosing the best car among a set of alternatives is a solution search problem. And the choice of a particular car does not have drastic consequences, and therefore, choice of car is a Lipschitz continuous problem. In this chapter, Lipschitz continuous problems are considered. Clearly, such a definition translates onto a condition that the function $f(\cdot)$ be Lipschitz continuous. Let its Lipschitz constant be L. Some common examples of Lipschitz continuous functions are polynomial functions, sine/cosine functions, and the absolute value function.

8.3.2 Collaborative Problem Solving

The collaborative problem solving approach where human and machine search for a solution together is as given below:

1. Initialize $i = 1$, $j = 1$, and set $\mathcal{S} = \mathcal{X}$ as the original search space.
2. If $f(p^{-1}(j)) \neq f^0$,
   set $\mathcal{S} = \mathcal{S} - \{\mathbf{x} : ||\mathbf{x} - p^{-1}(j)|| < \frac{1}{L}||f^0 - f(p^{-1}(j))||\}$, and $j = \max_{\mathbf{x}\in\mathcal{S}} p(\mathbf{x})$.
   Repeat Step 2.
3. $\mathbf{x}^0 = p^{-1}(j)$.

The basic idea is to remove *nearby* strategies from the search space, when a particular strategy fails. The above approach has the following contributions from human and machine:

- *Human contribution*: Knowledge of the preference function $p(\cdot)$ that decides the strategies to be evaluated.

- *Machine contribution*: Quick and accurate evaluation of strategies.
- *Common contribution*: Evaluation of set $\mathcal{S}$ after every step based on the Lipschitz constant L.

8.3.3 Analysis

Collaborative problem solving addresses the issues faced when only humans or only machines try to solve a problem. When only humans try to solve the problem, due to the complex nature of $f(\cdot)$, the evaluation of strategies at every step is complicated and might result in inaccurate and/or delayed evaluation. On the other hand, if only machines try to solve the problem, due to a lack of prior preference order $p(\cdot)$, they have to randomly determine the strategies to evaluate, which would require more number of iterations on an average to solve the problem. Therefore, such a collaborative approach is needed. Here, the number of iterations needed to find the solution using the collaborative problem solving approach is evaluated. Clearly, it depends on the preference order, the location of the true solution in the preference order, and the nature of the original problem. To determine the number of iterations needed to find the solution, define the neighborhood of a given strategy $\mathbf{x}$ as follows:

$$\mathcal{N}(\mathbf{x}) \triangleq \left\{ \mathbf{y} : ||\mathbf{y} - \mathbf{x}|| < \frac{1}{L} ||f(\mathbf{y}) - f(\mathbf{x})|| \right\}.$$

From a simple observation that is explained in further detail later, we get the following as the number of iterations needed to find the solution for $p(\mathbf{x}_0) > 2$:

$$\mathbb{N}(f, p) = p(\mathbf{x}_0) - \sum_{j=1}^{p(\mathbf{x}_0)-2} \mathbb{1} \left(\sum_{\forall 1 \le p_0 < p_1 < \cdots < p_j \le p(\mathbf{x}_0)-1} A^{p_0}_{p_1, p_2, \ldots, p_j} \right), \tag{8.33}$$

where $\mathbb{1}(\cdot)$ is the indicator function and

$$A^{p_0}_{p_1,p_2,\ldots,p_j} = \begin{cases} 1, & \text{if } p^{-1}(p_k) \in \mathcal{N}(p^{-1}(p_0)) \forall k = 1, \cdots, j, \\ 0, & \text{otherwise,} \end{cases} . \tag{8.34}$$

For $p(\mathbf{x}_0) = 1, 2$, we have $\mathbb{N}(f, p) = p(\mathbf{x}_0)$. Although the expression looks complicated, it is derived in a straightforward manner by observing that every time a solution is discarded, it also discards solutions in its neighborhood. Therefore, this might cause a reduction in the number of iterations if the discarded neighborhood consists of solution candidates with higher preference value than the true solution.

Note that the maximum number of iterations for the collaborative approach is $p(\mathbf{x}_0)$. For a machine that does not have the knowledge of the preference order, it randomly chooses the alternatives to be evaluated and would take more number of iterations on an average. On the other hand, although the human knows the preference order and would require same maximum number of iterations $p(\mathbf{x}_0)$, the computational time and accuracy are worse for the human than a machine. Therefore, a collaborative effort reduces the time and increases the accuracy of finding the solution. This provides a mathematical way of understanding the benefits of collaboration. One can determine when a collaborative effort is better than the individual problem-solving architectures. Consider the case when the computation time per iteration for a human is t_h and for a machine is t_m. Then, if the machine solves the problem on its own without the help of a machine, it would take a (best-case) total computational time of $T_M = p(\mathbf{x}_0)t_m$. On the other hand, a human alone would take $T_H = \mathbb{N}(f, p)t_h$. For the collaborative effort, the total time taken is $T_C = \mathbb{N}(f, p)t_m$, which is clearly better than both T_H and T_M, since $t_m < t_h$ and $\mathbb{N}(f, p) \leq p(\mathbf{x}_0)$. Similarly, an accuracy analysis can also be performed. Note that the solution search problem has only been introduced in this thesis. Several other interesting questions that arise from such a formulation will be addressed in the future.

8.4 Discussion

The knowledge discovery problem considered here is strongly related to the set of problems referred to as the coupon collector's problem [16, 23, 76]. Most of the results in the case of weighted coupon collectors problem (or coupon collector problem in general) consider the expected number of iterations required to collect all coupons while the knowledge discovery problem in this chapter addresses the opposite version where the average number of coupons collected after T iterations is evaluated. Also, most of the existing results of coupon collectors problem are approximations/asymptotic order results. Most importantly, as far as we know no results in the coupon collectors problem have been found for the noisy case or the case where each coupon is of different *quality* as considered here.

In this chapter, two problems have been explored where humans and machines can collaborate to improve the inference performance of tasks. Several intuitive observations have been made for the two problems considered. Although the problems explored are simple cases, it provides a mathematical understanding of the many benefits associated with using humans and machines together. They fall under the larger paradigm of Human-Machine Inference Networks (HuMaINs) which are of practical importance due to the technological advancements where humans and machines are supporting each other for various tasks. This framework needs to be further developed to accommodate a number of other tasks as discussed in the future work of the next chapter.

CHAPTER 9

CONCLUSION

9.1 Summary

In this thesis, the problem of accomplishing reliable inference from systems consisting of unreliable agents was addressed. The general methodology for this was to first analyze the effect of unreliable agents in the network and quantify their effect on the global performance of the network. The second step was to design schemes that are robust to such unreliable information from these agents. These schemes used coding-theoretic approaches to improve the individual performance of the agents and also correct the errors from them at the global agents. This analysis was performed for sensor networks first in Chapters 3 and 4 and then for human networks in Chapters 5–7. Finally, a human-machine collaborative framework was proposed in Chapter 8 to solve complex problems efficiently and quickly. Specifically, the contributions of this thesis are listed below.

In *Chapter 3*, the target localization and target tracking problems were investigated in a WSN under a Bayesian framework. A Monte Carlo based approach was developed for target localization. By assuming the target location to be random, the performance of a minimum mean square error (MMSE) estimator was analyzed in the presence of Byzantines by considering two kinds of attacks: independent attack and collaborative attacks. The appropriate performance metric for the Bayesian framework is Posterior Fisher Information or Posterior Cramér-Rao lower bound (PCRLB). The minimum fraction of Byzantines (α_{blind}) was defined as the fraction of Byzantines required to make the network non-informative to the FC. This was analytically derived by modeling the effect of Byzantines as a binary symmetric channel (BSC). Optimal strategies for both the Byzantines and the network were designed by modeling their behavior as a zero-sum game. PCRLB was used as the utility function and the Nash-Equilibrium was found as the saddle point. For the case of collaborative attacks, a lower bound on α_{blind} was found. Similarly, for the target tracking problem, α_{blind} has been found and the optimal attacking strategies have been found for the case when the fraction of Byzantines is lower than the blinding fraction.

In *Chapter 4*, techniques were investigated to make the network robust to the presence of Byzantines. Specifically, an adaptive learning scheme was proposed to identify the Byzantines in the network which was shown to be highly effective. Moreover, in order to improve the performance further, a dynamic iterative quantization scheme at the local sensors was proposed and derived using calculus of variations. The problem was formulated in a game-theoretic framework and the optimal quantizers for both the honest sensors and the Byzantines were derived. The proposed quantization scheme not only improved the estimation performance significantly but also made the Byzantines 'ineffective' when combined with the adaptive learning scheme. These two schemes have also been extended to target tracking problem. The third scheme proposed was different from the traditional optimal estimator at the FC and instead an asymptotically optimal and easy to implement scheme based on error-correcting codes was proposed. The fundamental theory was developed and asymptotic performance results were derived. The proposed scheme modeled the localization problem as a hierarchical classification problem. The scheme provided a coarse estimate in a computationally efficient manner as compared to the traditional ML based approach. The performance of the proposed scheme was determined in terms of detection probability of the correct region. It was analytically shown that the scheme achieves perfect performance in the asymptotic regime. The error correction capability of the coding theory based approach provides Byzantine tolerance capability and the use of soft-decoding at the FC provides tolerance to the channel errors. This shows the benefit of adopting coding theory based techniques for signal processing applications.

In *Chapter 5*, the CEO problem was considered for non-regular source distributions (such as uniform or truncated Gaussian). A group of agents observing independently corrupted versions of data, transmit coded versions over rate-limited links to a CEO. The CEO then estimates the underlying data based on the received coded observations. Agents are not allowed to convene before transmitting their observations. This formulation was motivated by the practical problem of a firm's CEO estimating (non-regular) beliefs about a sequence of events, before acting on them. Agents' observations were modeled as jointly distributed with the underlying data through a given conditional probability density function. The asymptotic behavior of the minimum achievable mean squared error distortion at the CEO was studied in the limit when the number of agents L and the sum rate R tend to infinity and established a $1/R^2$ convergence of the distortion.

In *Chapter 6*, the case of a human FC who fuses decisions from multiple humans was considered and the performance of decision fusion by people was compared to the optimal fusion rules. It was observed that the behavior is different since the optimal fusion rule is a deterministic one while people typically use non-deterministic rules which depend on various factors. Based on these observations, a hierarchical Bayesian model was developed to address the observed behavior of humans. This model captures the differences observed in people at individual level, crowd level, and population level. The effect of such models on the design of larger human-machine systems was demonstrated by designing hierarchical sociotechnical systems where the human decision fusion components in the system are modeled using the hierarchical Bayesian model and the machines in the system are optimized.

In *Chapter 7*, we focused on crowdsourcing for M-ary classification tasks, such as multi-class object recognition from images into fine-grained categories. Distributed classification codes and a minimum Hamming distance decoder were used to design the system in order to minimize misclassification probability such that workers need to only answer binary questions. The efficacy of this coding-based approach was demonstrated using simulations and through real data from Amazon Mechanical Turk, a paid crowdsourcing microtask platform. The approach was analyzed under different crowdsourcing models including the peer-dependent reward scheme and the dependent observations model. In the process, an ordering principle for the quality of crowds was also developed. For systems with peer-dependent reward schemes, it was observed that higher correlation among workers results in performance degradation. Further, if the workers also share dependent observations due to common sources of information, it was shown that the system performance deteriorates as expected. However, it was also observed that when the observations become independent, the performance gain due to the proposed coding-based approach over the majority-vote approach increases.

In *Chapter 8*, a general problem-solving architecture was considered and possible scopes of collaboration were outlined. Based on this architecture of human-machine inference networks, two example problems were considered. In the knowledge discovery problem, a human interested in discovering all the unknown elements of a set is supported by a machine. This partnership was referred to as machine as a coach collaboration. The performance of this learning process was characterized in terms of quantity and quality of the known elements at every time step. In the solution search problem, humans and machines collaborated as colleagues to determine the solution to a problem, such as finding a maximum point for a given function. The mathematical frameworks presented in this chapter provide an intuitive understanding of the benefits of human-machine collaboration and can help in the design of larger human-machine inference networks.

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. Later, more general future research is outlined.

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 [66]) 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*, the use of error-correcting codes was considered for a localization problem. However, some of the results were restrictive based on Assumption 4.4.3. In the future, one can extend this work by relaxing this assumption and to also derive the convergence rates using Berry-Essen inequalities. 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 5*, only the scaling behavior of quadratic non-regular CEO problem was considered. It is desired to derive precise characterizations for sum rate distortion and for full rate-distortion for this non-regular CEO problem as obtained by Oohama [102] and Prabhakaran, et al. [109], respectively for the quadratic Gaussian CEO problem. Similar to other CEO problems, one can observe the difference in decay rates for distortion between our result and the centralized case when agents can convene. When agents are allowed to convene, the setup is the single-terminal compression problem whose rate-distortion function under MSE was determined by Fix [49]. However, it has no simple expression and the optimizing solution has support on finite number of mass points. On the other hand, for absolute error distortion measure, rate-distortion function exists in closed form for uniform source [171] and it would be interesting to analyze the uniform CEO problem under the absolute error distortion. Gastpar and Eswaran [52] have addressed the CEO problem for non-Gaussian sources, but have considered the additive Gaussian noise channel. An interesting variant is when the source follows a regular distribution with a finite support and the measurement noise is modeled using copula. For example, beta distribution satisfies the regularity conditions and has a finite support. Also, for distributions such as cosine, parabolic, and inverted parabolic, midrange (similar to the one used in this chapter) is more efficient than mean [120]. In such cases, it will be interesting to explore if the minimum achievable square distortion would still exhibit a $1/R$ convergence behavior.

In *Chapter 6*, psychological principles were employed to understand the behavior of humans at fusing multiple decisions. The data needs to be analyzed with a fine-tooth comb to identify the individual cases when the decisions of humans do not match the CV rule's decision. A psychological understanding of these particular cases can help us in comprehending this complex phenomenon. Data should also be collected with a large number of sources (N) to verify some asymptotic approximations. In other words, this data should be used to verify the hypothesis that humans use heuristic decision rules when the amount of data is large. On similar lines, time-constrained tasks can also be considered, to verify if heuristic rules such as 'pick-the-best' rule which fail in the framework considered here, would work well under such time-constrained situations.

In *Chapter 7*, a crowdsourcing system was considered and a coding theory based scheme was designed to ensure reliable classification using unreliable crowds. Going forward, many further questions may be addressed; some examples are as follows. Can better cognitive and attentional models of human crowd workers provide better insight and design principles? When considering average misclassification probability, the ordering of crowd quality depends only on a first-moment characterization; what about finer characterizations of system performance? One can also design the number of paired workers in the peer-dependent reward scheme to optimize system performance. In the current work, we designed the code matrices with no prior knowledge of the crowd parameters. In the future, one can also consider a setup where the crowd parameters such as the covariance and/or dependence parameters are known to further improve the system performance.

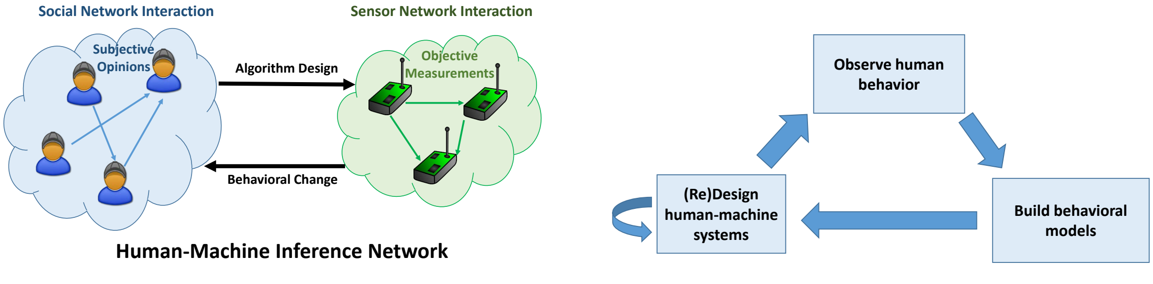
Fig. 9.1 summarizes future directions in terms of the general problems following this thesis and the general approach towards solving them. A HuMaIN as defined in Chapter 8 consists of a social network where humans exchange subjective opinions among themselves, and a sensor network where sensors exchange objective measurements amongst them. Moreover, due to the interaction between social and sensor networks, the behavioral characteristics of humans determine algorithms adopted by machines and these algorithms in turn affect the behavior of humans. Therefore, as suggested by the preliminary results, an intelligent collaboration of humans and machines can deliver improved results.

The future work can be summarized into two specific research directions:

1. Use of statistical modeling techniques to develop mathematical models of human decision making, in collaboration with cognitive psychologists, and

2. To use the above developed models to design robust fusion algorithms that handle unreliable data from the agents.

These projects have both theoretical and implementation challenges. Therefore, the focus is first on developing theoretical models for such a collaboration and then, implementing the designed algorithms to verify their applicability in practice. Both these problems are further explained in detail below.



(a) (b)

Fig. 9.1: Elements of the proposed research. (a) Unified architecture of human-machine inference networks (b) General approach for design and analysis of HuMaINs.

9.2.1 Statistical Modeling of Human Decision Making

The first step towards developing efficient systems containing humans and machines is to develop appropriate models characterizing their behavior. While statistical models exist that characterize the machine observations, researchers have not extensively investigated the modeling of decisions and subjective confidences on multihypothesis tasks, or on tasks in which human decision makers can provide imprecise (i.e., vague) decisions. Both of these task types, however, are important in the many applications of HuMaINs. In the preliminary work (Chapter 6), a comparative study between people and machines at the task of decision fusion has been performed. It was observed that the behavior is different since the optimal fusion rule is a deterministic one while people typically use non-deterministic rules which depend on various factors. Based on these observations, a hierarchical Bayesian model was developed to address the observed behavior of humans. This model captures the differences observed in people at individual level, crowd level, and population level. Moving forward, for individual human decision-making models, tools from bounded rationality framework [127] and rational inattention theory [128] can be used in building a theory. Experiments with human subjects can be designed to model the cognitive mechanisms which govern the generation of decisions and decision confidences as they pertain to the formulation of precise and imprecise decisions. One can also build models that consider the effect of stress, anxiety, and fatigue in the cognitive mechanisms of human decision making, decision confidence assessment, and response time (similar to [116, 168]).

9.2.2 Design of Robust Algorithms for Collaborative Decision Making

The next step after deriving probabilistic models of human decision-making, is to develop efficient fusion algorithms for collaborative decision making. The goal would be to seek optimal or near-optimal fusion rules which incorporate the informational nature of both humans and machines. Due to the large volume of data in some practical applications, it is also of interest to analyze the effects that a large number of agents (humans/machines) and a high rate of incoming data have on the performance of the fusion rules. However, the highly parameterized nature of these human models might deem their implementation impractical. Also the presence of unreliable components in the system might result in poor fusion performance. Data from existing studies in the cognitive psychology literature along with models resulting from the work in Sec. 9.2.1 can be used in the analysis of these operators. For cases in which the implementation of the optimal rule is not feasible, a future project can investigate the use of adaptive fusion rules that attempt to learn the parameters of the optimal fusion rule online. Also, for the design of simple and robust algorithms, ideas from coding theory can be used similar to the reliable crowdsourcing results previously derived in Chapter 7.

For the development of future systems consisting of humans and machines, the methodology described above needs to be implemented. First, statistical models of humans should be developed, which are then used to optimize the machines in the system. 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 unreliable agents.

APPENDIX A

APPENDIX

A.1 Proof of Proposition 4.3.1

The probability of a sensor sending a bit value 1 is

$$P(u=1|a) = Q\left(\frac{\eta(\hat{a})-a}{\sigma}\right). \tag{A.1}$$

The data's contribution to the posterior Fisher Information is given by

$$F = -E\Big[\frac{\partial^2 \ln P(u|a)}{\partial^2 a}\Big], \tag{A.2}$$

where $\ln p(u|a) = (1-u)\ln(1-P(u=1|a)) + u\ln P(u=1|a)$. Let $P_1 = P(u=1|a)$ and $P_0 = 1-P_1$. Then

$$\frac{\partial^2 \ln p(u|a)}{\partial^2 a} = -\frac{(1-u)}{P_0^2}\Big(\frac{\partial P_0}{\partial a}\Big)^2 + \frac{(1-u)}{P_0}\frac{\partial^2 P_0}{\partial^2 a} - \frac{u}{P_1^2}\Big(\frac{\partial P_1}{\partial a}\Big)^2 + \frac{u}{P_1}\frac{\partial^2 P_1}{\partial^2 a}$$

and

$$E\Big[\frac{\partial^2 \ln p(u|a)}{\partial^2 a}\Big] = -\frac{1}{P_0}\Big(\frac{\partial P_0}{\partial a}\Big)^2 - \frac{1}{P_1}\Big(\frac{\partial P_1}{\partial a}\Big)^2 \tag{A.3}$$

Note the fact that $E[u] = P_1$ has been used in (A.3). Since $P_1 = 1 - P_0$,

$$\Big(\frac{\partial P_0}{\partial a}\Big)^2 = \Big(\frac{\partial P_1}{\partial a}\Big)^2 = \frac{e^{-\frac{(\eta(\hat{a})-a)^2}{\sigma^2}}}{2\pi\sigma^2}, \tag{A.4}$$

where the relation

$$\frac{\partial Q[(\frac{\eta(\hat{a})-a}{\sigma})]}{\partial a} = \frac{e^{-\frac{(\eta(\hat{a})-a)^2}{2\sigma^2}}}{\sigma\sqrt{2\pi}} \tag{A.5}$$

has been used. From (A.2), (A.3) and (A.4), we get the desired result.

A.2 Proof of Proposition 4.3.3

Let the probability of a sensor sending a bit value 1 be defined as P_1.

$$\begin{aligned} P_1 &= P(u=1|a, Byzantine)P(Byzantine) + P(u=1|a, Honest)P(Honest) \\ &= \alpha\left(p\left(1 - Q\left(\frac{\eta^B(\hat{a}) - a}{\sigma}\right)\right) + (1-p)Q\left(\frac{\eta^B(\hat{a}) - a}{\sigma}\right)\right) + \\ &\qquad (1-\alpha)Q\left(\frac{\eta^H(\hat{a}) - a}{\sigma}\right), \end{aligned} \tag{A.6}$$

where $P(Byzantine) = \alpha$ is the probability that the sensor is a Byzantine, and similarly, $P(Honest) = 1 - \alpha$ is the probability that the sensor is honest. Here, p denotes the probability of flipping by the Byzantines. The data's contribution to the posterior Fisher Information is given by

$$F = -E\Big[\frac{\partial^2 \ln P(u|a)}{\partial^2 a}\Big], \tag{A.7}$$

where $\ln p(u|a) = (1-u)\ln(1-P_1) + u \ln P_1$. Let $P_0 = 1 - P_1$. Then

$$\frac{\partial^2 \ln p(u|a)}{\partial^2 a} = -\frac{(1-u)}{P_0^2}\Big(\frac{\partial P_0}{\partial a}\Big)^2 + \frac{(1-u)}{P_0}\frac{\partial^2 P_0}{\partial^2 a} - \frac{u}{P_1^2}\Big(\frac{\partial P_1}{\partial a}\Big)^2 + \frac{u}{P_1}\frac{\partial^2 P_1}{\partial^2 a}$$

and

$$E\Big[\frac{\partial^2 \ln p(u|a)}{\partial^2 a}\Big] = -\frac{1}{P_0}\Big(\frac{\partial P_0}{\partial a}\Big)^2 - \frac{1}{P_1}\Big(\frac{\partial P_1}{\partial a}\Big)^2 \tag{A.8}$$

Note the fact that $E[u] = P_1$ has been used in (A.8). Since $P_1 = 1 - P_0$,

$$\Big(\frac{\partial P_0}{\partial a}\Big)^2 = \Big(\frac{\partial P_1}{\partial a}\Big)^2 = \frac{\Bigg(-\alpha(2p-1)e^{-\frac{(\eta_B(\hat{a})-a)^2}{2\sigma^2}} + (1-\alpha)e^{-\frac{(\eta_H(\hat{a})-a)^2}{2\sigma^2}}\Bigg)^2}{2\pi\sigma^2}, \tag{A.9}$$

where the relation

$$\frac{\partial Q[(\frac{\eta(\hat{a})-a}{\sigma})]}{\partial a} = \frac{e^{-\frac{(\eta(\hat{a})-a)^2}{2\sigma^2}}}{\sigma\sqrt{2\pi}} \tag{A.10}$$

has been used. From (A.7), (A.8) and (A.9), we get the desired result.

Note that when $\alpha = 0$, the data's contribution to posterior Fisher Information in (4.14) becomes

$$F[\eta(\hat{a}), a] = \frac{e^{-\frac{(\eta(\hat{a})-a)^2}{\sigma^2}}}{2\pi\sigma^2[Q(\frac{\eta(\hat{a})-a}{\sigma})][1 - Q(\frac{\eta(\hat{a})-a}{\sigma})]}, \tag{A.11}$$

which is the result of Proposition 4.3.1.

A.3 Proof of Lemma 4.4.2

Let $d_H(\cdot,\cdot)$ be the Hamming distance between two vectors, for fixed $\theta \in R_j^k$,

$$\begin{aligned}
P_e^k(\theta) &= P\left\{\text{detected region} \neq R_j^k|\theta\right\} \\
&\leq P\left\{d_H(\boldsymbol{u}^k, \boldsymbol{c}_{j+1}^k) \geq \min_{0\leq l\leq M-1, l\neq j} d_H(\boldsymbol{u}^k, \boldsymbol{c}_{l+1}^k)|\theta\right\} \\
&\leq \sum_{0\leq l\leq M-1, l\neq j} P\left\{d_H(\boldsymbol{u}^k, \boldsymbol{c}_{j+1}^k) \geq d_H(\boldsymbol{u}^k, \boldsymbol{c}_{l+1}^k)|\theta\right\} \\
&= \sum_{0\leq l\leq M-1, l\neq j} P\left\{\sum_{\{i\in[1,\cdots,N_k]:c_{(l+1)i}\neq c_{(j+1)i}\}} z_{i,j}^k \geq 0|\theta\right\}.
\end{aligned} \tag{A.12}$$

Using the fact that $c_{(l+1)i}^k \neq c_{(j+1)i}^k$ for all $i \in S_j^k \cup S_l^k$, $l \neq j$, we can simplify the above equation. Also, observe that $\{z_{i,j}\}_{i=1}^{N_k}$ are independent across the sensors given θ. According to (2) in [170],

$$\lambda_m = \frac{1}{d_{m,k}} \sum_{i=1}^{N_k} (c^k_{(l+1)i} \oplus c^k_{(j+1)i})(2q^k_{i,j} - 1) = \frac{1}{d_{m,k}} \sum_{i \in S^k_j \cup S^k_l} (2q^k_{i,j} - 1) = \frac{1}{d_{m,k}} \left(\sum_{i \in S^k_j \cup S^k_l} 2q^k_{i,j} - \frac{2N_k}{M} \right) \tag{A.13}$$

since $c^k_{(l+1)i} \neq c^k_{(j+1)i}$ for all $i \in S^k_j \cup S^k_l$, $l \neq j$. $\lambda_m < 0$ is then equivalent to condition (4.33). Therefore, using Lemma 4.4.1 and (A.13),

$$P\left\{ \sum_{\{i \in [1,\cdots,N_k]: c_{(l+1)i} \neq c_{(j+1)i}\}} z^k_{i,j} \geq 0 | \theta \right\} \leq \left(1 - \frac{\left(\sum_{i \in S^k_j \cup S^k_l} (2q^k_{i,j} - 1) \right)^2}{d^2_{m,k}} \right)^{d_{m,k}/2} \tag{A.14}$$

Substituting (A.14) into (A.12), we have (4.34). Note that condition (4.33) ($\lambda_m < 0$) implies $\lambda^k_{j,\max}(\theta) < 0$ by definition. Hence, (4.35) is a direct consequence from (4.34).

A.4 Proof of Theorem 4.4.4

First we prove that condition (4.33) is satisfied by the proposed scheme for all θ when $\sigma < \infty$. Hence, the inequality (4.35) can be applied to the proposed scheme. The probabilities $q^k_{i,j}$ given by (4.37) are

$$q^k_{i,j} = \begin{cases} 1 - Q\left(\frac{(\eta^k_i - a_i)}{\sigma}\right), & \text{for } i \in S^k_j \\ Q\left(\frac{(\eta^k_i - a_i)}{\sigma}\right), & \text{for } i \in S^k_l \end{cases}. \tag{A.15}$$

By Assumption 4.4.3, there exists a bijection function f from S^k_j to S^k_l. The sum $\sum_{i \in S^k_j \cup S^k_l} q^k_{i,j}$ of (4.33) can be evaluated by considering pairwise summations as follows. Let us consider one such pair $(i_j \in S_j^k, f(i_j) = i_l \in S_l^k)$. Hence, their thresholds are $\eta_{i_j}^k = \eta_{i_l}^k = \eta$. Then, from (A.15),

$$\begin{aligned} q_{i_j,j}^k + q_{i_l,j}^k &= 1 - Q\left(\frac{(\eta - a_{i_j})}{\sigma}\right) + Q\left(\frac{(\eta - a_{i_l})}{\sigma}\right) && \text{(A.16)} \\ &= 1 - \left[Q\left(\frac{(\eta - a_{i_j})}{\sigma}\right) - Q\left(\frac{(\eta - a_{i_l})}{\sigma}\right)\right]. && \text{(A.17)} \end{aligned}$$

Now observe that, by the assumption,

$$a_{i_j} = \frac{\sqrt{P_0}}{d_{i_j}} > \frac{\sqrt{P_0}}{d_{i_l}} = a_{i_l}$$

and, therefore, $Q\left(\frac{(\eta - a_{i_j})}{\sigma}\right) > Q\left(\frac{(\eta - a_{i_l})}{\sigma}\right)$ for all finite values of σ. From (A.17), the sum $q_{i_j,j}^k + q_{i_l,j}^k$ is strictly less than 1. Therefore, the sum $\sum_{i \in S_j^k \cup S_l^k} q_{i,j}^k < \frac{N_k}{M} = \frac{N}{M^{k+1}} = \frac{d_{m,k}}{2}$. Therefore, the condition in (4.33) is satisfied for the code matrix used in this scheme. Hence, $P_e^k(\theta)$ can always be bounded by (4.35).

By using (4.35), P_d^k can be bounded as follows:

$$\begin{aligned} P_d^k &= 1 - \sum_{j=0}^{M-1} P\{\theta \in R_j^k\} P\left\{\text{detected region} \neq R_j^k | \theta \in R_j^k\right\} \\ &= 1 - \frac{1}{M} \sum_{j=0}^{M-1} \int_\theta P\{\theta | \theta \in R_j^k\} P\left\{\text{detected region} \neq R_j^k | \theta, \theta \in R_j^k\right\} \, d\theta \\ &= 1 - \frac{1}{M} \sum_{j=0}^{M-1} \int_{\theta \in R_j^k} P\{\theta | \theta \in R_j^k\} P_e^k(\theta) \, d\theta \\ &\geq 1 - \frac{1}{M} \sum_{j=0}^{M-1} \int_{\theta \in R_j^k} P\{\theta | \theta \in R_j^k\} (M-1) \left(1 - \left(\lambda_{j,\max}^k(\theta)\right)^2\right)^{d_{m,k}/2} \, d\theta \\ &\geq 1 - \frac{M-1}{M} \sum_{j=0}^{M-1} \left(1 - \left(\lambda_{j,\max}^k\right)^2\right)^{d_{m,k}/2} \int_{\theta \in R_j^k} P\{\theta | \theta \in R_j^k\} \, d\theta && \text{(A.18)} \\ &\geq 1 - \frac{M-1}{M} \sum_{j=0}^{M-1} \left(1 - \left(\lambda_{\max}^k\right)^2\right)^{d_{m,k}/2} && \text{(A.19)} \\ &= 1 - (M-1) \left(1 - \left(\lambda_{\max}^k\right)^2\right)^{d_{m,k}/2}. && \text{(A.20)} \end{aligned}$$

Both (A.18) and (A.19) are true since $\lambda_{j,\max}^k < 0$ and $\lambda_{\max}^k < 0$.

A.5 Proof of Theorem 4.6.3

First we prove that when $\alpha < 0.5$, then

$$\sum_{i \in S_j^k \cup S_l^k} Z_i^{jl} E[\psi_i^k | \theta] \to \infty, \tag{A.21}$$

where $Z_i^{jl} = \frac{1}{2}((-1)^{c_{ji}^k} - (-1)^{c_{li}^k})$. Based on our code matrix design, Z_i^{jl} for $i \in S_j^k \cup S_l^k$ is given as

$$Z_i^{jl} = \begin{cases} -1, & \text{for } i \in S_j^k \\ +1, & \text{for } i \in S_l^k \end{cases}. \tag{A.22}$$

By using the pairwise summation approach discussed in Section 4.4.2, notice that, for every sensor $i_j \in S_j^k$ and its corresponding sensor $i_l \in S_l^k$, when $\theta \in R_j^k$,

$$Z_{i_j}^{jl} E[\psi_{i_j}^k | \theta] + Z_{i_l}^{jl} E[\psi_{i_l}^k | \theta] \quad = \quad E[(\psi_{i_l}^k - \psi_{i_j}^k) | \theta]. \tag{A.23}$$

Now, for a given sensor i,

$$\begin{aligned} E[\psi_i^k | \theta] &= P(u_i^k = 0 | \theta) E[\psi_i^k | \theta, u_i^k = 0] + P(u_i^k = 1 | \theta) E[\psi_i^k | \theta, u_i^k = 1] && \text{(A.24)} \\ &= (1 - P(u_i^k = 1 | \theta)) E[\psi_i^k | u_i^k = 0] + P(u_i^k = 1 | \theta) E[\psi_i^k | u_i^k = 1] && \text{(A.25)} \\ &= E[\psi_i^k | u_i^k = 0] + P(u_i^k = 1 | \theta) \left[E[\psi_i^k | u_i^k = 1] - E[\psi_i^k | u_i^k = 0] \right], && \text{(A.26)} \end{aligned}$$

where the facts that $P(u_i^k = 0 | \theta) + P(u_i^k = 1 | \theta) = 1$ and that the value of ψ_i^k depends only on u_i^k have been used.

Note that the channel statistics are the same for both the sensors. Therefore, $E[\psi_i^k | u_i^k = d]$ for

$d = \{0, 1\}$ given by

$$E[\psi_i^k | u_i^k = d] \;\; = \;\; E\left[\ln \frac{P(v_i^k | u_i^k = 0)}{P(v_i^k | u_i^k = 1} \middle| u_i^k = d\right]$$

is the same for both the sensors. The pairwise sum $E[(\psi_{i_l}^k - \psi_{i_j}^k)|\theta]$ now simplifies to the following,

$$\begin{aligned}
& E[(\psi_{i_l}^k - \psi_{i_j}^k)|\theta] \\
= \;\; & E[\psi_i^k | u_i^k = 0] + P(u_{i_l}^k = 1|\theta)\left[E[\psi_i^k | u_i^k = 1] - E[\psi_i^k | u_i^k = 0]\right] \\
- \;\; & E[\psi_i^k | u_i^k = 0] - P(u_{i_j}^k = 1|\theta)\left[E[\psi_i^k | u_i^k = 1] - E[\psi_i^k | u_i^k = 0]\right] && \text{(A.27)} \\
= \;\; & \Big(P(u_{i_l}^k = 1|\theta) - P(u_{i_j}^k = 1|\theta)\Big)\left[E[\psi_i^k | u_i^k = 1] - E[\psi_i^k | u_i^k = 0]\right]. && \text{(A.28)}
\end{aligned}$$

When $\theta \in R_j^k$,

$$\begin{aligned}
P(u_{i_j}^k = 1|\theta) \;\; &= \;\; \alpha + (1 - 2\alpha) Q\left(\frac{(\eta - a_{i_j})}{\sigma}\right) && \text{(A.29)} \\
P(u_{i_l}^k = 1|\theta) \;\; &= \;\; \alpha + (1 - 2\alpha) Q\left(\frac{(\eta - a_{i_l})}{\sigma}\right) && \text{(A.30)}
\end{aligned}$$

and, therefore,

$$P(u_{i_l}^k = 1|\theta) - P(u_{i_j}^k = 1|\theta) = (1 - 2\alpha)\left(Q\left(\frac{(\eta - a_{i_l})}{\sigma}\right) - Q\left(\frac{(\eta - a_{i_j})}{\sigma}\right)\right). \qquad \text{(A.31)}$$

Note that, since $\theta \in R_j^k$, $Q\left(\frac{(\eta - a_{i_l})}{\sigma}\right) < Q\left(\frac{(\eta - a_{i_j})}{\sigma}\right)$. Next we prove that

$$E[\psi_i^k | u_i^k = 1] - E[\psi_i^k | u_i^k = 0] < 0 \qquad \text{(A.32)}$$

for all finite noise variance of the fading channel (σ_f^2).

$$\begin{aligned}
& E[\psi_i^k|u_i^k=1] - E[\psi_i^k|u_i^k=0] \\
= \;& E\left[\ln\frac{P(v_i^k|u_i^k=0)}{P(v_i^k|u_i^k=1}\middle| u_i^k=1\right] - E\left[\ln\frac{P(v_i^k|u_i^k=0)}{P(v_i^k|u_i^k=1}\middle| u_i^k=0\right] \\
= \;& \int_{-\infty}^{\infty} P(v_i^k|u_i^k=1)\ln\frac{P(v_i^k|u_i^k=0)}{P(v_i^k|u_i^k=1}\, dv_i^k - \int_{-\infty}^{\infty} P(v_i^k|u_i^k=0)\ln\frac{P(v_i^k|u_i^k=0)}{P(v_i^k|u_i^k=1}\, dv_i^k \\
= \;& -\left[D(P(v_i^k|u_i^k=1)||P(v_i^k|u_i^k=0) + D(P(v_i^k|u_i^k=0)||P(v_i^k|u_i^k=1)\right],
\end{aligned} \tag{A.33}$$

where $D(p||q)$ is the Kullback-Leiber distance between probability distributions p and q. Since $P(v_i^k|u_i^k=1) \neq P(v_i^k|u_i^k=0)$ for all finite σ_f^2, we have $D(P(v_i^k|u_i^k=1)||P(v_i^k|u_i^k=0) > 0$ and $D(P(v_i^k|u_i^k=0)||P(v_i^k|u_i^k=1) > 0$. This concludes that $E[\psi_i^k|u_i^k=1] - E[\psi_i^k|u_i^k=0] < 0$. Hence, When $\alpha < 1/2$, from (A.28), (A.31), and (A.32), $E[(\psi_{i_l}^k - \psi_{i_j}^k)|\theta] > 0$ and the condition $\sum_{i\in S_j^k\cup S_l^k} Z_i^{jl} E[\psi_i^k|\theta] \to \infty$ is satisfied.

We now show that when the condition (A.21) is satisfied, the proposed scheme asymptotically attains perfect detection probability.

$$\begin{aligned}
\lim_{N\to\infty} P_D &= \lim_{N\to\infty} \prod_{k=0}^{k^{stop}} P_d^k \\
&\geq \prod_{k=0}^{k^{stop}} \lim_{N\to\infty} \left[1 - \sum_{j=0}^{M-1} P\left\{\theta \in R_j^k\right\} P\left\{\text{detected region} \neq R_j^k | \theta \in R_j^k\right\}\right] \\
&= \prod_{k=0}^{k^{stop}} \lim_{N\to\infty} \left[1 - \frac{1}{M}\sum_{j=0}^{M-1} \int_\theta P\left\{\theta | \theta \in R_j^k\right\} P\left\{\text{detected region} \neq R_j^k | \theta, \theta \in R_j^k\right\} d\theta\right].
\end{aligned}$$

Define

$$P_{e,j,\max}^k \triangleq \max_{\theta\in R_j^k} P_{e,j}^k(\theta) \tag{A.34}$$

and

$$P_{e,\max}^k \triangleq \max_{0\leq j\leq M-1} P_{e,j,\max}^k. \tag{A.35}$$

Then,

$$\begin{aligned}
\lim_{N\to\infty} P_D &= \prod_{k=0}^{k^{stop}} \lim_{N\to\infty}\left[1-\frac{1}{M}\sum_{j=0}^{M-1}\int_{\theta} P\left\{\theta|\theta\in R_j^k\right\} P_{e,j}^k(\theta)d\theta\right] \\
&\geq \prod_{k=0}^{k^{stop}} \lim_{N\to\infty}\left[1-\frac{1}{M}\sum_{j=0}^{M-1}\int_{\theta\in R_j^k} P\left\{\theta|\theta\in R_j^k\right\} P_{e,j,\max}^k d\theta\right] \\
&= \prod_{k=0}^{k^{stop}} \lim_{N\to\infty}\left[1-\frac{1}{M}\sum_{j=0}^{M-1} P_{e,j,\max}^k \int_{\theta\in R_j^k} P\left\{\theta|\theta\in R_j^k\right\} d\theta\right] \\
&\geq \prod_{k=0}^{k^{stop}} \lim_{N\to\infty}\left[1-\frac{P_{e,\max}^k}{M}\sum_{j=0}^{M-1} 1\right] \\
&= \prod_{k=0}^{k^{stop}} \left[1-\lim_{N\to\infty} P_{e,\max}^k\right]. \qquad \text{(A.36)}
\end{aligned}$$

Since $E\left[(\tilde{\psi}_i^k)^2|\theta\right]$ is bounded as shown by Lemma 4.6.2, Lindeberg condition [48] holds and $\frac{1}{\sigma_{\tilde{\psi}}(\theta)}\sum_{i\in S_j^k\cup S_l^k} Z_i^{jl}\tilde{\psi}_i^k$ tends to a standard Gaussian random variable by Lindeberg central limit theorem [48]. Therefore, from (4.51),

$$\begin{aligned}
\lim_{N\to\infty} P_{e,j}^k(\theta) &\leq \lim_{N\to\infty} \sum_{0\leq l\leq M-1, l\neq j} P\left\{\frac{1}{\sigma_{\tilde{\psi}}(\theta)}\sum_{i\in S_j^k\cup S_l^k} Z_i^{jl}\tilde{\psi}_i^k < -\frac{1}{\sigma_{\tilde{\psi}}(\theta)}\sum_{i\in S_j^k\cup S_l^k} Z_i^{jl} E[\psi_i^k|\theta]\,\Big|\,\theta\right\} \qquad \text{(A.37)} \\
&= \sum_{0\leq l\leq M-1, l\neq j} \lim_{N\to\infty} Q\left(\frac{1}{\sigma_{\tilde{\psi}}(\theta)}\sum_{i\in S_j^k\cup S_l^k} Z_i^{jl} E[\psi_i^k|H_j^k]\right). \qquad \text{(A.38)}
\end{aligned}$$

Since, for a fixed θ, $\sigma_{\tilde{\psi}}(\theta)$ will grow slower than $\sum_{i\in S_j^k\cup S_l^k} Z_i^{jl} E[\psi_i^k|\theta]$ when $\sum_{i\in S_j^k\cup S_l^k} Z_i^{jl} E[\psi_i^k|\theta] \to \infty$, $\lim_{N\to\infty} P_{e,j}^k(\theta) = 0$ for all θ. Hence, $\lim_{N\to\infty} P_{e,\max}^k = 0$ and from (A.36), $\lim_{N\to\infty} P_D = 1$ for all finite noise variance.

A.6 Proposition A.6.1

Proposition A.6.1.

$$K^2E\left[\left(\frac{\hat{U}_{(1)}+\hat{U}_{(L)}}{2}-\frac{a+b}{2}\right)^2\right] \;\leq\; 2K^2E\left[\left(\frac{U_{(1)}+U_{(L)}}{2}-\frac{a+b}{2}\right)^2\right]+\epsilon_3$$

where $\epsilon_3=\epsilon_3(\delta_0,n)$ *can be made arbitrarily small by making* n *sufficiently large and* δ_0 *sufficiently small.*

Proof. Let $\mathcal{B}$ be the event $\{\tilde{U}_i\neq\hat{U}_i,\forall i\}$. By inequality (5.20), we have $\Pr\{\mathcal{B}\}\leq\lambda$. Now,

$$\begin{aligned}
&K^2E\Bigg[\left(\frac{\hat{U}_{(1)}+\hat{U}_{(L)}}{2}-\frac{a+b}{2}\right)^2-2\left(\frac{U_{(1)}+U_{(L)}}{2}-\frac{a+b}{2}\right)^2\Bigg]\\
=&\frac{K^2}{4}E\Bigg[\left(\hat{U}_{(1)}+\hat{U}_{(L)}-(a+b)\right)^2-2\left(U_{(1)}+U_{(L)}-(a+b)\right)^2\Bigg]\\
=&\frac{K^2}{4}E\Bigg[\Big((\hat{U}_{(1)}+\hat{U}_{(L)})-(U_{(1)}+U_{(L)})-\big((a+b)-(U_{(1)}+U_{(L)})\big)\Big)^2-2\left(U_{(1)}+U_{(L)}-(a+b)\right)^2\Bigg]\\
\leq&\frac{K^2}{4}E\Bigg[2\left((\hat{U}_{(1)}+\hat{U}_{(L)})-(U_{(1)}+U_{(L)})\right)^2+2\left((a+b)-(U_{(1)}+U_{(L)})\right)^2\\
&-2\left(U_{(1)}+U_{(L)}-(a+b)\right)^2\Bigg]\\
=&\frac{K^2}{2}E\left[\left((\hat{U}_{(1)}+\hat{U}_{(L)})-(U_{(1)}+U_{(L)})\right)^2\right]\\
\leq&K^2E\left[\left((U_{(1)}+U_{(L)})-(\tilde{U}_{(1)}+\tilde{U}_{(L)})\right)^2\right]+K^2E\left[\left((\tilde{U}_{(1)}+\tilde{U}_{(L)})-(\hat{U}_{(1)}+\hat{U}_{(L)})\right)^2\right]\\
\leq&2K^2E\left[\left(U_{(1)}-\tilde{U}_{(1)}\right)^2\right]+2K^2E\left[\left(U_{(L)}-\tilde{U}_{(L)}\right)^2\right]+2K^2E\left[\left(\tilde{U}_{(1)}-\hat{U}_{(1)}\right)^2\right]\\
&+2K^2E\left[\left(\tilde{U}_{(L)}-\hat{U}_{(L)}\right)^2\right],
\end{aligned}$$

where $\tilde{U}_{(i)}$ are the order statistics of $\tilde{U}_i$, and the last two inequalities follow from the fact that $E[(A+B)^2]\leq 2E[A^2]+2E[B^2]$.

Now, choose δ_0 to be sufficiently small to ensure that the ordering of variates U_i is preserved under quantization. Then, $U_{(1)}$ and $\tilde{U}_{(1)}$ correspond to the same agent's data, say the ℓth agent. Therefore,

$$E\left[\left(U_{(1)} - \tilde{U}_{(1)}\right)^2\right] = E\left[\left(U_\ell - \tilde{U}_\ell\right)^2\right] \leq \delta_0$$

by (5.15). Similarly, $E\left[\left(U_{(L)} - \tilde{U}_{(L)}\right)^2\right] \leq \delta_0$. Also, define $\tilde{u}_{\text{max}} = \max\{|\tilde{u}| : \tilde{u} \in \tilde{\mathcal{U}}\}$. Now, for $i = \{1, \ldots, L\}$:

$$\begin{aligned}
E\left[\left(\tilde{U}_{(i)} - \hat{U}_{(i)}\right)^2\right] &= \sum_{u,u'} \left(\tilde{U}_{(i)} - \hat{U}_{(i)}\right)^2 \Pr\left\{\tilde{U}_{(i)} = u, \hat{U}_{(i)} = u'\right\} \\
&= \sum_{u,u'} \left(\tilde{U}_{(i)} - \hat{U}_{(i)}\right)^2 \Pr\left\{\tilde{U}_{(i)} = u, \hat{U}_{(i)} = u'\right\} \\
&\leq \sum_{u,u'} 4\tilde{u}_{\text{max}}^2 \Pr\left\{\tilde{U}_{(i)} = u, \hat{U}_{(i)} = u'\right\} \\
&= 4\tilde{u}_{\text{max}}^2 \Pr\left\{\tilde{U}_{(i)} \neq \hat{U}_{(i)}\right\} \\
&\leq 4\tilde{u}_{\text{max}}^2 \Pr\left\{\mathcal{B}\right\} \\
&\leq 4\tilde{u}_{\text{max}}^2 \lambda.
\end{aligned}$$

Therefore,

$$\begin{aligned}
K^2 E\left[\left(\frac{\hat{U}_{(1)} + \hat{U}_{(L)}}{2} - \frac{a+b}{2}\right)^2 - 2\left(\frac{U_{(1)} + U_{(L)}}{2} - \frac{a+b}{2}\right)^2\right] &\leq 4K^2\delta_0 + 8K^2\tilde{u}_{\text{max}}^2 Pr(\mathcal{B}) \\
&\leq 4K^2\delta_0 + 8K^2\tilde{u}_{\text{max}}^2 \lambda.
\end{aligned}$$

Now, choosing a sufficiently large n such that

$$\lambda < \frac{\epsilon_3 - 4K^2\delta_0}{8K^2\tilde{u}_{\text{max}}^2},$$

yields the desired result. □

A.7 Proposition A.7.1

Proposition A.7.1. *The following inequality:*

$$\frac{\sum_{i=1}^{n} p_i A_i}{\sum_{i=1}^{n} p_i B_i} \geq \min_i \left(\frac{A_i}{B_i} \right) \tag{A.39}$$

holds, if $p_i, A_i, B_i \geq 0$ *and not all are* 0*.*

Proof. Let $m = \min_i \left(\frac{A_i}{B_i} \right)$. By definition,

$$\begin{aligned} A_i &\geq B_i m, \text{ for all } i = 1, \ldots, n \\ \Longrightarrow p_i A_i &\geq p_i B_i m, \text{ for all } i = 1, \ldots, n \\ \Longrightarrow \sum_{i=1}^{n} p_i A_i &\geq m \sum_{i=1}^{n} p_i B_i \\ \Longrightarrow \frac{\sum_{i=1}^{n} p_i A_i}{\sum_{i=1}^{n} p_i B_i} &\geq m = \min_i \left(\frac{A_i}{B_i} \right). \end{aligned}$$

□

REFERENCES

[1] H. Akçay, H. Hjalmarsson, and L. Ljung, "On the choice of norms in system identification," *IEEE Trans. Autom. Control*, vol. 41, no. 9, pp. 1367–1372, Sep. 1996. 108, 110

[2] I. F. Akyildiz, W. Su, Y. Sankarasubramaniam, and E. Cayirci, "A survey on sensor networks," *IEEE Commun. Mag.*, vol. 40, no. 8, pp. 102–114, Aug. 2002. 1

[3] 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. 18

[4] J. R. Anderson, *Cognitive Psychology and Its Implications*. New York: Worth Publishers, 2010. 164

[5] 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. 18

[6] ——, "Decentralized detection with censoring sensors," *IEEE Trans. Signal Process.*, vol. 56, no. 4, pp. 1362–1373, Apr. 2008. 18

[7] S. Arbesman, "Quantifying the ease of scientific discovery," *Scientometrics*, vol. 86, no. 2, pp. 245–250, Jun. 2011. 169, 178, 179

[8] S. Arbesman and N. A. Christakis, "Eurekometrics: Analyzing the nature of discovery," *PLoS Comput. Biol.*, vol. 7, no. 6, pp. 1–2, Jun. 2011. 169, 178

[9] S. Arbesman and G. Laughlin, “A scientometric prediction of the discovery of the first potentially habitable planet with a mass similar to earth,” *PLoS ONE*, vol. 5, no. 10, pp. 1–4, Oct. 2010. 169, 178

[10] G. R. Arce and S. A. Fontana, “On the midrange estimator,” *IEEE Trans. Acoust., Speech, Signal Process.*, vol. 36, no. 6, pp. 920–922, Jun. 1988. 101, 107

[11] M. S. Arulampalam, S. Maskell, N. Gordon, and T. Clapp, “A tutorial on particle filters for online nonlinear/non-Gaussian Bayesian tracking,” *IEEE Trans. Signal Process.*, vol. 50, no. 2, pp. 174–188, Feb. 2002. 43

[12] F. Aurenhammer, “Voronoi diagram– A survey of a fundamental geometric data structure,” *ACM Comput. Surv.*, vol. 23, no. 3, pp. 345–405, Sep. 1991. 68

[13] Y. Bar-Shalom, X.-R. Li, and T. Kirubarajan, *Estimation with Applications to Tracking and Navigation*. New York: John Wiley & Sons, 2001. 41

[14] K. L. Bell, “Performance bounds in parameter estimation with application to bearing estimation,” Ph.D. dissertation, George Mason University, 1995. 112

[15] K. L. Bell, Y. Steinberg, Y. Ephraim, and H. L. Van Trees, “Extended Ziv-Zakai lower bound for vector parameter estimation,” *IEEE Trans. Inf. Theory*, vol. 43, no. 2, pp. 624–637, Mar. 1997. 112

[16] P. Berenbrink and T. Sauerwald, “The weighted coupon collector’s problem and applications,” in *Computing and Combinatorics*, H. Q. Ngo, Ed. Berlin-Heidelberg: Springer, 2009, pp. 449–458. 184

[17] T. Berger, Z. Zhang, and H. Viswanathan, “The CEO problem,” *IEEE Trans. Inf. Theory*, vol. 42, no. 3, pp. 887–902, May 1996. 94, 98, 112, 118

[18] P. Berkhin, “Survey of clustering data mining techniques,” Accrue Software, Inc., San Jose, CA, USA, Tech. Rep., 2002. 68

[19] L. Berti-Equille, A. D. Sarma, X. Dong, A. Marian, and D. Srivastava, "Sailing the information ocean with awareness of currents: Discovery and application of source dependence," in *Proc. Conf. Innov. Data Syst. Research*, Jan. 2009. 133

[20] 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. 1, 18

[21] M. A. Boden, *The Creative Mind: Myths and Mechanisms*. New York: Routledge, 2004. 167

[22] D. Bollier, *The Future of Work: What It Means for Individuals, Businesses, Markets and Governments*. Washington, DC: The Aspen Institute, 2011. 3, 4

[23] A. Boneh and M. Hofri, "The coupon-collector problem revisited- a survey of engineering problems and computational methods," *Stochastic Models*, vol. 13, no. 1, pp. 39–66, Jan. 1997. 184

[24] A. Boukerche, H. A. B. F. Oliveira, E. F. Nakamura, and A. A. F. Loureiro, "Secure localization algorithms for wireless sensor networks," *IEEE Commun. Mag.*, vol. 46, no. 4, pp. 96–101, Apr. 2008. 4

[25] S. Branson, C. Wah, F. Schroff, B. Babenko, P. Welinder, P. Perona, and S. Belongie, "Visual recognition with humans in the loop," in *Computer Vision – ECCV 2010*, ser. Lecture Notes in Computer Science, L. Buttyán, V. Gligor, and D. Westhoff, Eds. Berlin: Springer, 2010, vol. 6314, pp. 438–451. 5, 10

[26] Z. Chair and P. K. Varshney, "Optimal data fusion in multiple sensor detection systems," *IEEE Trans. Aerosp. Electron. Syst.*, vol. AES-22, no. 1, pp. 98–101, Jan. 1986. 121, 126

[27] 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. 18

[28] ——, “Wireless sensors in distributed detection applications,” *IEEE Signal Process. Mag.*, vol. 24, no. 3, pp. 16–25, May 2007. 18

[29] D. Chazan, M. Zakai, and J. Ziv, “Improved lower bounds on signal parameter estimation,” *IEEE Trans. Inf. Theory*, vol. IT-21, no. 1, pp. 90–93, Jan. 1975. 112

[30] 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. 4, 5, 18

[31] H. Chen, “Noise enhanced signal detection and estimation,” Ph.D. dissertation, Syracuse University, 2007. 174

[32] H. Chen, L. R. Varshney, and P. K. Varshney, “Noise-enhanced information systems,” *Proc. IEEE*, vol. 102, no. 10, pp. 4667–471, Oct. 2014. 174

[33] 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

[34] J. C. Chen, R. E. Hudson, and K. Yao, “A maximum likelihood parametric approach to source localization,” in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process. (ICASSP 2001)*, May 2001, pp. 3013–3016. 2

[35] J. Chen and J. Wang, “On the vector Gaussian CEO problem,” in *Proc. 2011 IEEE Int. Symp. Inf. Theory (ISIT)*, Jul. 2011, pp. 2050–2054. 94

[36] J. Chen, X. Zhang, T. Berger, and S. B. Wicker, “An upper bound on the sum-rate distortion function and its corresponding rate allocation schemes for the CEO problem,” *IEEE J. Sel. Areas Commun.*, vol. 22, no. 6, pp. 977–987, Aug. 2004. 94

[37] R. Chen, J.-M. Park, and K. Bian, “Robust distributed spectrum sensing in cognitive radio networks,” in *Proc. 27th IEEE Conf. Computer Commun. (INFOCOM 2008)*, Apr. 2008, pp. 1876–1884. 20

[38] U. Cherubini, E. Luciano, and W. Vecchiato, *Copula Methods in Finance*. New York: Wiley, 2004. 3

[39] T. M. Cover, "A proof of the data compression theorem of Slepian and Wolf for ergodic sources," *IEEE Trans. Inf. Theory*, vol. IT-21, no. 2, pp. 226–228, Mar. 1975. 101

[40] T. M. Cover and J. A. Thomas, *Elements of Information Theory*. New York: John Wiley & Sons, 1991. 116

[41] H. Cramér, *Mathematical Methods of Statistics*. Princeton, NJ: Princeton Univ. Press, 1946. 107

[42] 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. 18

[43] H. A. David and H. N. Nagaraja, *Order Statistics*, 3rd ed. Hoboken, NJ: Wiley-Interscience, 2003. 107

[44] R. L. Dobrushin, "General formulation of Shannon's main theorem of information theory," *Usp. Math. Nauk.*, vol. 14, no. 6(90), pp. 3–104, 1959, translated in *Am. Math. Soc. Trans.*, vol. 33, pp. 323-348. 102

[45] A. Doucet, N. de Freitas, and N. Gordon, *Sequential Monte Carlo Methods in Practice*. New York: Springer, 2001. 43

[46] A. Doucet and X. Wang, "Monte Carlo methods for signal processing: A review in the statistical signal processing context," *IEEE Signal Process. Mag.*, vol. 22, no. 6, pp. 152–170, Nov. 2005. 28

[47] J. S. Downs, M. B. Holbrook, S. Sheng, and L. F. Cranor, "Are your participants gaming the system? screening Mechanical Turk workers," in *Proc. 28th SIGCHI Conf. Hum. Factors Comput. Syst. (CHI 2010)*, Apr. 2010, pp. 2399–2402. 5

[48] W. Feller, *An Introduction to Probability Theory and its Applications*. New York, NY: Wiley, 1966. 202

[49] S. L. Fix, "Rate distortion functions for squared error distortion measures," in *Proc. 16th Annu. Allerton Conf. Commun. Control Comput.*, Oct. 1978, pp. 704–711. 100, 189

[50] D. Fudenberg and J. Tirole, *Game Theory*. Cambridge, MA: MIT Press, 1991. 33, 34

[51] 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. 20

[52] M. Gastpar and K. Eswaran, "On the quadratic AWGN CEO problem and non-Gaussian sources," in *Proc. 2005 IEEE Int. Symp. Inf. Theory (ISIT)*, Sep. 2005, pp. 219–223. 94, 189

[53] C. Genest and J. MacKay, "The joy of copulas: Bivariate distributions with uniform marginals," *Am. Stat.*, vol. 40, no. 4, pp. 280–283, Nov. 1986. 3

[54] G. Gigerenzer and R. Selten, *Bounded Rationality: The Adaptive Toolbox*. Cambridge: MIT Press, 2002. 3, 119

[55] D. A. Grier, "Error identification and correction in human computation: Lessons from the WPA," in *Proc. AAAI Workshop Human Comput. (HCOMP'11)*, Aug. 2011, pp. 32–36. 133

[56] T. L. Griffiths, C. Kemp, and J. B. Tenenbaum, "Bayesian models of cognition," in *The Cambridge Handbook of Computational Cognitive Modeling*, R. Sun, Ed. New York: Cambridge University Press, 2008, pp. 59–100. 3

[57] T. S. Han and S.-I. Amari, "Statistical inference under multiterminal data compression," *IEEE Trans. Inf. Theory*, vol. 44, no. 6, pp. 2300–2324, Oct. 1998. 95

[58] 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. 20

[59] J. Henrich, S. J. Heine, and A. Norenzayan, "Most people are not WEIRD," *Nature*, vol. 466, no. 7302, p. 29, Jul. 2010. 125

[60] C. Hildreth, "Bayesian statisticians and remote clients," *Econometrica*, vol. 31, no. 3, pp. 422–438, Jul. 1963. 95

[61] S.-W. Huang and W.-T. Fu, "Don't hide in the crowd! increasing social transparency between peer workers improves crowdsourcing outcomes," in *Proc. SIGCHI Conf. Hum. Factors Comput. Syst. (CHI 2013)*, Apr. 2013, pp. 621–630. 10, 133, 149

[62] P. G. Ipeirotis, F. Provost, and J. Wang, "Quality management on Amazon Mechanical Turk," in *Proc. ACM SIGKDD Workshop Human Comput. (HCOMP'10)*, Jul. 2010, pp. 64–67. 5, 133

[63] 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. 18

[64] 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. 18

[65] 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. 20

[66] 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. 189

[67] ——, "Asymptotic analysis of distributed Bayesian detection with Byzantine data," *IEEE Signal Process. Lett.*, vol. 22, no. 5, pp. 608–612, May 2015. 20

[68] L. M. Kaplan, Q. Li, and P. Molnar, “Maximum likelihood methods for bearings-only target localization,” in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process. (ICASSP 2001)*, May 2001, pp. 3001–3004. 2

[69] 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. 18

[70] D. R. Karger, S. Oh, and D. Shah, “Budget-optimal crowdsourcing using low-rank matrix approximations,” in *Proc. 49th Annu. Allerton Conf. Commun. Control Comput.*, Sep. 2011, pp. 284–291. 133

[71] ——, “Iterative learning for reliable crowdsourcing systems,” in *Advances in Neural Information Processing Systems 24*, J. Shawe-Taylor, R. Zemel, P. Bartlett, F. Pereira, and K. Weinberger, Eds. Cambridge, MA: MIT Press, 2011, pp. 1953–1961. 133

[72] ——, “Efficient crowdsourcing for multi-class labeling,” in *Proc. ACM SIGMETRICS Int. Conf. Meas. Model. Comput. Syst.*, Jun. 2013, pp. 81–92. 134

[73] S. M. Kay, *Fundamentals of Statistical Signal Processing: Estimation Theory*. Upper Saddle River, NJ: Prentice Hall PTR, 1993. 16

[74] ——, *Fundamentals of Statistical Signal Processing: Detection Theory*. Upper Saddle River, NJ: Prentice Hall PTR, 1998. 16

[75] R. D. King, J. Rowland, W. Aubrey, M. Liakata, M. Markham, L. N. Soldatova, K. E. Whelan, A. Clare, M. Young, A. Sparkes, S. G. Oliver, and P. Pir, “The robot scientist Adam,” *Computer*, vol. 42, no. 8, pp. 46–54, Aug. 2009. 167

[76] J. E. Kobza, S. H. Jacobson, and D. E. Vaughan, “A survey of the coupon collector’s problem with random sample sizes,” *Methodol. Comput. Appl. Prob.*, vol. 9, no. 4, pp. 573–584, Dec. 2007. 184

[77] O. Kosut and L. Tong, "Distributed source coding in the presence of Byzantine sensors," *IEEE Trans. Inf. Theory*, vol. 54, no. 6, pp. 2550–2565, Jun. 2008. 25

[78] O. Kosut, L. Tong, and D. Tse, "Nonlinear network coding is necessary to combat general Byzantine attacks," in *Proc. 47th Annu. Allerton Conf. Commun. Control Comput.*, Oct. 2009, pp. 593–599. 25

[79] G. Kramer and S. A. Savari, "Communicating probability distributions," *IEEE Trans. Inf. Theory*, vol. 53, no. 2, pp. 518–525, Feb. 2007. 95

[80] 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. 18

[81] L. Lamport, R. Shostak, and M. Pease, "The Byzantine generals problem," *IEEE Trans. Autom. Control*, vol. 4, no. 3, pp. 382–401, Jul. 1982. 19

[82] P. Langley, "The computational support of scientific discovery," *Int. J. Human-Computer Studies*, vol. 53, no. 3, pp. 393–410, Sep. 2000. 167

[83] E. Law and L. von Ahn, *Human Computation*. Morgan & Claypool Publishers, 2011. 132

[84] M. Lease, "On quality control and machine learning in crowdsourcing," in *Proc. AAAI Workshop Human Comput. (HCOMP'11)*, Aug. 2011, pp. 97–102. 5

[85] D. Li and Y. H. Hu, "Energy-based collaborative source localization using acoustic microsensor array," *EURASIP J. Appl. Signal Process.*, vol. 2003, no. 4, pp. 331–337, Apr. 2003. 2, 27

[86] Z. Li, W. Trappe, Y. Zhang, and B. Nath, "Robust statistical methods for securing wireless localization in sensor networks," in *Proc. 4th Int. Symp. Inf. Processing Sensor Netw. (IPSN'05)*, Apr. 2005, pp. 9–98. 4

[87] B. Liu and B. Chen, "Joint source-channel coding for distributed sensor networks," in *Conf. Rec. 38th Asilomar Conf. Signals, Syst. Comput.*, Nov. 2004, pp. 1397–1401. 5

[88] T. Lubart, “How can computers be partners in the creative process: Classification and commentary on the special issue,” *Int. J. Human-Computer Studies*, vol. 63, no. 4–5, pp. 365–369, Oct. 2005. 166

[89] 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. 18

[90] 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. 20, 25

[91] S. Marano, V. Matta, P. Willett, and L. Tong, “DOA estimation via a network of dumb sensors under the SENMA paradigm,” *IEEE Signal Process. Lett.*, vol. 12, no. 10, pp. 709–712, Oct. 2005. 2

[92] ——, “Support-based and ML approaches to DOA estimation in a dumb sensor network,” *IEEE Trans. Signal Process.*, vol. 54, no. 4, pp. 1563–1567, Apr. 2006. 2

[93] J. G. March and H. A. Simon, *Organizations*. New York: Wiley, 1958. 165

[94] J. Marschak and R. Radner, *Economic Theory of Teams*. New Haven: Yale University Press, 1972. 4, 5, 125

[95] E. Masazade, R. Niu, and P. K. Varshney, “Energy aware iterative source localization schemes for wireless sensor networks,” *IEEE Trans. Signal Process.*, vol. 58, no. 9, pp. 4824–4835, Sep. 2010. 28, 51

[96] W. Mason and D. J. Watts, “Financial incentives and the “performance of crowds”,” in *Proc. ACM SIGKDD Workshop Human Comput. (HCOMP’09)*, Jun. 2009, pp. 77–85. 132

[97] C. Meesookho, U. Mitra, and S. Narayanan, “On energy-based acoustic source localization for sensor networks,” *IEEE Trans. Signal Process.*, vol. 56, no. 1, pp. 365–377, Jan. 2008. 27

[98] 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. 20

[99] R. B. Nelsen, *An Introduction to Copulas*. New York: Springer, 2006. 3, 99

[100] J. Neyman and E. S. Pearson, “On the use and interpretation of certain test criteria for purposes of statistical inference: Part I,” *Biometrika*, vol. 20A, no. 1/2, pp. 175–240, Jul. 1928. 101, 107

[101] 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. 1, 2, 4, 31, 56, 57, 63, 66, 82

[102] Y. Oohama, “The rate-distortion function for the quadratic Gaussian CEO problem,” *IEEE Trans. Inf. Theory*, vol. 44, no. 3, pp. 1057–1070, May 1998. 94, 189

[103] ——, “Rate-distortion theory for Gaussian multiterminal source coding systems with several side informations at the decoder,” *IEEE Trans. Inf. Theory*, vol. 51, no. 7, pp. 2577–2593, Jul. 2005. 94

[104] O. Ozdemir, R. Niu, and P. K. Varshney, “Adaptive local quantizer design for tracking in a wireless sensor network,” in *Conf. Rec. 42nd Asilomar Conf. Signals, Syst. Comput.*, Oct. 2008, pp. 1202–1206. 63, 64

[105] ——, “Channel aware target localization with quantized data in wireless sensor networks,” *IEEE Trans. Signal Process.*, vol. 57, no. 3, pp. 1190–1202, Mar. 2009. 18, 32

[106] N. Patwari and A. O. Hero III, “Using proximity and quantized RSS for sensor localization in wireless networks,” in *Proc. 2nd Int. ACM Workshop Wireless Sens. Netw. Appl. 2003*, Sep. 2003, pp. 20–29. 2

[107] J. W. Payne and J. R. Bettman, “Walking with the scarecrow: The information-processing approach to decision research,” in *Blackwell Handbook of Judgment and Decision Making*, D. J. Koehler and N. Harvey, Eds. Oxford, UK: Blackwell Publishing Ltd., 2004, pp. 110–132. 131

[108] G. J. Pottie and W. J. Kaiser, “Wireless integrated network sensors,” *Commun. ACM*, vol. 43, no. 5, pp. 52–58, May 2000. 4

[109] V. Prabhakaran, D. Tse, and K. Ramachandran, “Rate region of the quadratic Gaussian CEO problem,” in *Proc. 2004 IEEE Int. Symp. Inf. Theory (ISIT)*, June-July 2004, p. 117. 94, 189

[110] J. B. Predd, R. Seiringer, E. H. Lieb, D. N. Osherson, H. V. Poor, and S. R. Kulkarni, “Probabilistic coherence and proper scoring rules,” *IEEE Trans. Inf. Theory*, vol. 55, no. 10, pp. 4786–4792, Oct. 2009. 3, 94

[111] G.-J. Qi, C. C. Aggarwal, J. Han, and T. Huang, “Mining collective intelligence in diverse groups,” in *Proc. 22nd Int. Conf. World Wide Web (WWW’13)*, May 2013, pp. 1041–1052. 10, 133, 155, 156

[112] A. J. Quinn and B. B. Bederson, “Human computation: a survey and taxonomy of a growing field,” in *Proc. 2011 Annu. Conf. Hum. Factors Comput. Syst. (CHI 2011)*, May 2011, pp. 1403–1412. 132

[113] R. Radner, “Team decision problems,” *Ann. Math. Stat.*, vol. 33, no. 3, pp. 857–881, Sep. 1962. 15

[114] 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. 18

[115] K. Ratakonda, R. Williams, J. Bisceglia, R. W. Taylor, and J. Graham, “Identifying trouble patterns in complex IT services engagements,” *IBM J. Res. Develop.*, vol. 54, no. 2, p. 5, Mar.-Apr. 2010. 3, 93

[116] R. Ratcliff and H. P. A. van Dongen, “Diffusion model for one-choice reaction-time tasks and the cognitive effects of sleep deprivation,” *Proc. Natl. Acad. Sci. U.S.A.*, vol. 108, no. 27, pp. 11 285–11 290, Jul. 2011. 192

[117] A. Rawat, P. Anand, H. Chen, and P. K. 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. 20, 25, 40

[118] J. B. Rhim, L. R. Varshney, and V. K. Goyal, “Quantization of prior probabilities for collaborative distributed hypothesis testing,” *IEEE Trans. Signal Process.*, vol. 60, no. 9, pp. 4537–4550, Sep. 2012. 95

[119] A. Ribeiro and G. B. 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. 18, 27, 59

[120] P. R. Rider, “The midrange of a sample as an estimator of the population midrange,” *J. Am. Stat. Assoc.*, vol. 52, no. 280, pp. 537–542, Dec. 1957. 107, 190

[121] J. Rocker, C. M. Yauch, S. Yenduri, L. A. Perkins, and F. Zand, “Paper-based dichotomous key to computer based application for biological identification,” *J. Comput. Sci. Coll.*, vol. 22, no. 5, pp. 30–38, May 2007. 136

[122] J. Ross, L. Irani, M. S. Silberman, A. Zaldivar, and B. Tomlinson, “Who are the crowdworkers? shifting demographics in Mechanical Turk,” in *Proc. 28th SIGCHI Conf. Hum. Factors Comput. Syst. (CHI 2010)*, Apr. 2010, pp. 2863–2872. 5

[123] R. K. Sawyer, *Explaining Creativity: The Science of Human Innovation*. Oxford: Oxford University Press, 2012. 167

[124] S. D. Servetto, "Achievable rates for multiterminal source coding with scalar quantizers," in *Conf. Rec. 39th Asilomar Conf. Signals, Syst. Comput.*, Oct. 2005, pp. 1762–1766. 101

[125] C. E. Shannon, *Reliable Machines from Unreliable Components*, MIT, Cambridge, MA, Mar. 1956, notes of first five lectures in the seminar of information theory. 6

[126] X. Shen and Y.-H. Hu, "Maximum likelihood multiple-source localization using acoustic energy measurements with wireless sensor network," *IEEE Trans. Signal Process.*, vol. 53, no. 1, pp. 44–53, Jan. 2005. 2, 27

[127] H. A. Simon, *Models of Bounded Rationality: Empirically Grounded Economic Reason*. Cambridge: MIT Press, 1982. 165, 192

[128] C. A. Sims, "Implications of rational inattention," *J. Monet. Econ.*, vol. 50, no. 3, pp. 665–690, Apr. 2003. 192

[129] D. Slepian and J. K. Wolf, "Noiseless coding of correlated information sources," *IEEE Trans. Inf. Theory*, vol. IT-19, no. 4, pp. 471–480, Jul. 1973. 101, 104

[130] R. Snow, B. O'Connor, D. Jurafsky, and A. Y. Ng, "Cheap and fast—but is it good?: Evaluating non-expert annotations for natural language tasks," in *Proc. Conf. Empirical Meth. Natural Language Process. (EMNLP'08)*, Oct. 2008, pp. 254–263. 10, 145

[131] J. B. Soll and R. P. Larrick, "Strategies for revising judgment: How (and how well) people use others' opinion," *J. Exp. Psychol.*, vol. 35, no. 3, pp. 780–805, May 2009. 3

[132] R. D. Sorkin, C. J. Hays, and R. West, "Signal-detection analysis of group decision making," *Psychol. Rev.*, vol. 108, no. 1, pp. 183–203, Jan. 2001. 3

[133] R. Soundararajan, A. B. Wagner, and S. Vishwanath, "Sum rate of the vacationing-CEO problem," *IEEE Trans. Inf. Theory*, vol. 58, no. 10, pp. 6304–6319, Oct. 2012. 94

[134] 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. 18

[135] 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. 18

[136] D. Tapscott and A. D. Williams, *Wikinomics: How Mass Collaboration Changes Everything*. New York: Portfolio Penguin, 2006. 3, 4

[137] ——, *Macrowikinomics: Rebooting Business and the World*. New York: Portfolio Penguin, 2010. 3, 4

[138] S. Tavildar and P. Viswanath, “On the sum-rate of the vector Gaussian CEO problem,” in *Conf. Rec. 39th Asilomar Conf. Signals, Syst. Comput.*, Oct. 2005, pp. 3–7. 94

[139] W. P. Tay, J. N. Tsitsiklis, and M. Z. Win, “Data fusion trees for detection: Does architecture matter?” *IEEE Trans. Inf. Theory*, vol. 54, no. 9, pp. 4155–4168, Sep. 2008. 18

[140] M. G. Taylor, “Reliable information storage in memories designed from unreliable components,” *Bell Syst. Tech. J.*, vol. 47, no. 10, pp. 2229–2337, Dec. 1968. 6

[141] 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. 18

[142] C. Thompson, *Smarter Than You Think: How Technology is Changing Our Minds for the Better*. New York: Penguin Press, 2013. 180

[143] P. Tichavský, C. H. Muravchik, and A. Nehorai, “Posterior Cramér-Rao bounds for discrete-time nonlinear filtering,” *IEEE Trans. Signal Process.*, vol. 46, no. 2, pp. 1386–1395, May 1998. 43

[144] R. E. Valdés-Pérez, “Principles of human–computer collaboration for knowledge discovery in science,” *Artif. Intell.*, vol. 107, no. 2, pp. 335–346, Feb. 1999. 167

[145] H. L. Van Trees, *Detection, Estimation and Modulation Theory*. New York, NY: Wiley, 1968, vol. 1. 9, 29, 112

[146] H. L. Van Trees and K. L. Bell, *Bayesian Bounds for Parameter Estimation and Nonlinear Filtering/Tracking*. Hoboken, NJ: Wiley-IEEE, 2007. 29

[147] B. vanBrunt, *The Calculus of Variations*. New York, NY: Springer, 2004. 59

[148] K. R. Varshney and L. R. Varshney, “Quantization of prior probabilities for hypothesis testing,” *IEEE Trans. Signal Process.*, vol. 56, no. 10, pp. 4553–4562, Oct. 2008. 95

[149] ——, “Optimal grouping for group minimax hypothesis testing,” *IEEE Trans. Inf. Theory*, vol. 60, no. 10, pp. 6511–6521, Oct. 2014. 95

[150] L. R. Varshney, “Unreliable and resource-constrained decoding,” Ph.D. dissertation, Massachusetts Institute of Technology, 2010. 6

[151] ——, “Participation in crowd systems,” in *Proc. 50th Annu. Allerton Conf. Commun. Control Comput.*, Oct. 2012, pp. 996–1001. 132

[152] ——, “Privacy and reliability in crowdsourcing service delivery,” in *Proc. SRII Global Conf. 2012*, Jul. 2012, pp. 55–60. 5

[153] L. R. Varshney, A. Vempaty, and P. K. Varshney, “Assuring privacy and reliability in crowdsourcing with coding,” in *Proc. 2014 Inf. Theory Appl. Workshop*, Feb. 2014, pp. 1–6. 5

[154] P. K. Varshney, *Distributed Detection and Data Fusion*. New York: Springer-Verlag, 1996. 1, 5, 18, 125

[155] 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. 1, 15, 17

[156] A. Vempaty, K. Agrawal, H. Chen, and P. K. Varshney, “Adaptive learning of Byzantines’ behavior in cooperative spectrum sensing,” in *Proc. 2011 IEEE Wireless Commun. Netw. Conf. (WCNC)*, Mar. 2011, pp. 1310–1315. 20, 49, 51, 52

[157] 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. 18

[158] 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. 20

[159] 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. 4, 20

[160] 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. 18

[161] P. Viswanath, “Sum rate of a class of Gaussian multiterminal source coding problems,” in *Advances in Network Information Theory*, P. Gupta, G. Kramer, and A. J. van Wijngaarden, Eds. Providence: DIMACS, American Mathematical Society, 2004, pp. 43–64. 94

[162] H. Viswanathan and T. Berger, “The quadratic Gaussian CEO problem,” *IEEE Trans. Inf. Theory*, vol. 43, no. 5, pp. 1549–1559, Sep. 1997. 9, 10, 94, 98, 101, 103, 104, 106, 112, 113, 118

[163] 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. 1, 18

[164] A. B. Wagner, S. Tavildar, and P. Viswanath, “Rate region of the quadratic Gaussian two-encoder source-coding problem,” *IEEE Trans. Inf. Theory*, vol. 54, no. 5, pp. 1938–1961, Sep. 2008. 94, 101

[165] T.-Y. Wang, Y. S. Han, B. Chen, and P. K. Varshney, “A combined decision fusion and channel coding scheme for distributed fault-tolerant classification in wireless sensors networks,” *IEEE Trans. Wireless Commun.*, vol. 5, no. 7, pp. 1695–1705, Jul. 2006. 22, 86

[166] T.-Y. Wang, Y. S. Han, P. K. Varshney, and P.-N. Chen, “Distributed fault-tolerant classification in wireless sensors networks,” *IEEE J. Sel. Areas Commun.*, vol. 23, no. 4, pp. 724–734, Apr. 2005. 21, 22, 133, 143, 148

[167] X. Wang, M. Fu, and H. Zhang, “Target tracking in wireless sensor networks based on the combination of KF and MLE using distance measurements,” *IEEE Trans. Mobile Comput.*, vol. 11, no. 4, pp. 567–576, Apr. 2012. 1

[168] C. N. White, R. Ratcliff, M. W. Vasey, and G. McKoon, “Anxiety enhances threat processing without competition among multiple inputs: A diffusion model analysis,” *Emotion*, vol. 10, no. 5, pp. 662–677, Oct. 2010. 192

[169] Y. Yang and Z. Xiong, “On the generalized Gaussian CEO problem,” *IEEE Trans. Inf. Theory*, vol. 58, no. 6, pp. 3350–3372, Jun. 2012. 94

[170] C. Yao, P.-N. Chen, T.-Y. Wang, Y. S. Han, and P. K. Varshney, “Performance analysis and code design for minimum Hamming distance fusion in wireless sensor networks,” *IEEE Trans. Inf. Theory*, vol. 53, no. 5, pp. 1716–1734, May 2007. 73, 133, 139, 197

[171] K. Yao and H. H. Tan, “Absolute error rate-distortion functions for sources with constrained magnitude,” *IEEE Trans. Inf. Theory*, vol. IT-24, no. 4, pp. 499–503, Jul. 1978. 189

[172] R. Zamir and T. Berger, “Mutliterminal source coding with high resolution,” *IEEE Trans. Inf. Theory*, vol. 45, no. 1, pp. 106–117, Jan. 1999. 101

[173] Y. Zhang, W. Liu, and Y. Fang, “Secure localization in wireless sensor networks,” in *Proc. 2005 IEEE Military Commun. Conf. (MILCOM)*, Oct. 2005, pp. 3169–3175. 4

[174] J. Ziv and M. Zakai, “Some lower bounds on signal parameter estimation,” *IEEE Trans. Inf. Theory*, vol. IT-15, no. 3, pp. 386–391, May 1969. 112

VITA

NAME OF AUTHOR: Aditya Vempaty

PLACE OF BIRTH: Warangal, Andhra Pradesh, India

DATE OF BIRTH: August 3, 1989

GRADUATE AND UNDERGRADUATE SCHOOLS ATTENDED:

Indian Institute of Technology, Kanpur, India

DEGREES AWARDED: B. Tech, 2011, Indian Institute of Technology, Kanpur, India